

30 Años de Calidad no son Nada...

CSTIC
1983 - 2013

personas



procesos



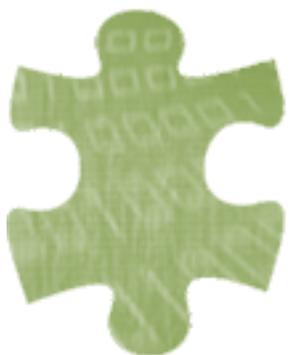
tecnologías



30 Años de Calidad no son Nada...

Comité de Calidad en los Sistemas y Tecnologías de la Información y la Comunicación

personas



procesos

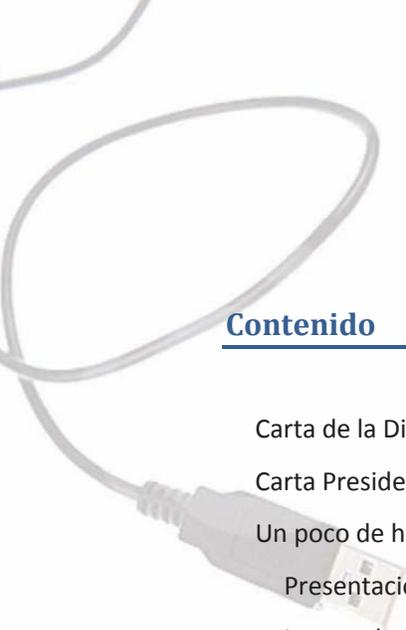


tecnologías

30 Aniversario

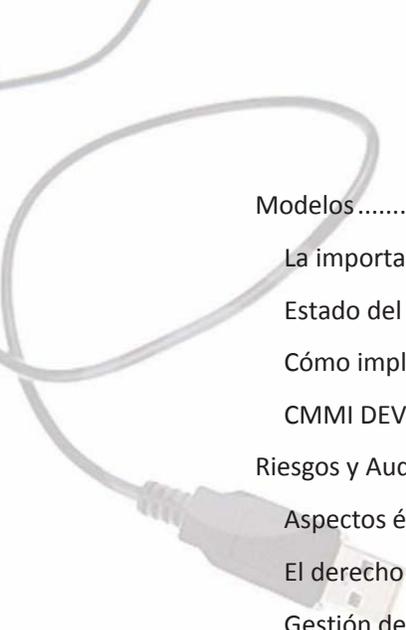
1982 - 2012

CSTIC

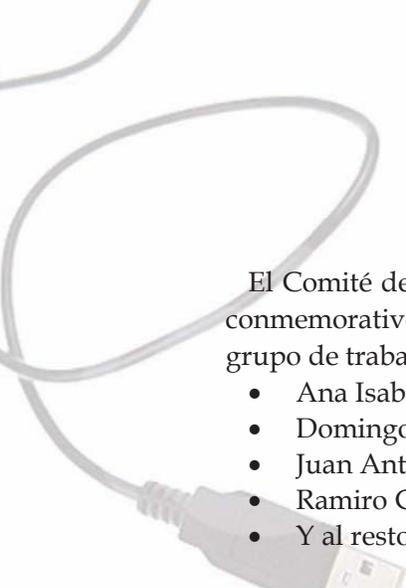


Contenido

Carta de la Dirección de la AEC.....	7
Carta Presidente del Comité.....	9
Un poco de historia.....	11
Presentación del Comité de Calidad de Software de la A.E.C.C.	15
Los productos Software y su Calidad hoy.....	21
Aspectos metodológicos y normativos en el Control de Calidad del Software.....	35
La Formación: Una clave esencial en la Mejora de la Calidad del Software.....	45
I Congreso Ibérico de la Calidad en el Software y Sistemas de Información.....	55
Las empresas de Tecnologías de la información y la nueva economía	57
Paradigma de Calidad	65
Nuevos retos, viejos problemas El camino hacia una cultura de ‘Excelencia en el Desarrollo de Software’	69
Gobernador de TI, Nuevo garante de la calidad de las TIC dentro de la Organización	73
La descentralización de los proyectos de Software.....	77
Medidas	81
Modelos para estimaciones del tamaño del software en proyectos.	85
Evaluación de proveedores mediante la gestión de un repositorio de proyectos con una herramienta de estimación	91
El papel de la técnica de Puntos Función en el ‘Outsourcing’ del Desarrollo de Aplicaciones	101
Evolución de los planes de Métricas para el control de procesos de desarrollo de software desarrollo de software	107
Indicadores de Calidad del Software: el valor de los números.	117
Medición Práctica de Software y Sistemas (PSM) Información objetiva para la toma de decisiones	125
Testing	131
Pruebas automatizadas y calidad para aplicaciones y sitios web.....	135
Validación del producto software	141
Prevención de Defectos: un modelo de análisis de productos potencialmente “fétidos” (stinkers)	145
La especialización de las pruebas del software.....	151



Modelos	155
La importancia de la certificación: De ISO a SW-CMM.....	159
Estado del Arte ISO/IEC 15504 Estado del Arte en la ISO/IEC 15504.	161
Cómo implantar CMMI y no morir en el intento	169
CMMI DEV Diferencias entre nivel 2 y nivel 3 y una estrategia de implantación	175
Riesgos y Auditorías.....	181
Aspectos éticos en las Auditorías de Calidad de Software	183
El derecho a crear Gestión de Riesgos en Proyectos Software El derecho a crear	187
Gestión de la incertidumbre en el proceso de estimación durante el ciclo de vida	197
Esta web ha sido atacada	205
Casos prácticos	211
La certificación CMMI-DEV, nivel 3 de unos Juegos Olímpicos	213
Procesos y aspectos clave en el sector de los productos sanitarios.....	219
Nuevos retos.....	229
Optimización energética en el ámbito de las TIC, Nueva cita en la agenda corporativa	231
Una aproximación a los costes para conseguir la presencia en redes sociales.....	239
Las redes sociales profesionales y el desarrollo de estándares en la industria del ‘software’ y la accesibilidad de Internet	243
La revolución social en las empresas.....	247
Premios Calidad CSTIC	253
Premio Calidad CSTIC 2009.....	253
Premios Calidad CSTIC 2011	255
Empresas y miembros del comité.....	261



El Comité de Calidad en los Sistemas y Tecnologías de la Información ha editado este libro conmemorativo de sus 30 años de vigencia. Queremos agradecer la dedicación de tiempo del grupo de trabajo que ha participado en la preparación de la publicación:

- Ana Isabel González Fernández
- Domingo Manuel Gaitero Gordillo
- Juan Antonio Caloto Caloto
- Ramiro Carballo Gutiérrez
- Y al resto de vocales por sus aportes

Por otro lado queremos mostrar nuestro agradecimiento a la entidad que ha permitido editar la publicación en papel a:

Proceso2ocial

Pasión por la tecnología

www.procesosocial.com

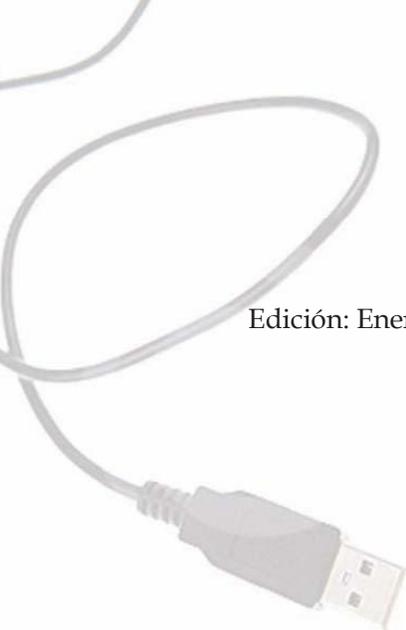
Este libro refleja parte de las actividades que el comité de Software, posteriormente renombrado como Comité de Calidad en los Sistemas y Tecnologías de la Información y la Comunicación ha llevado a cabo durante sus 30 años de existencia. Debido a esto en el texto aparecen información sobre empresas, personas, cargos y situaciones, que son de la época en la que fueron escritos los diferentes artículos, por lo que en la actualidad estos datos pueden ser diferentes.

Todas las demás marcas registradas que se mencionan, usan o citan en el presente documento son propiedad de los respectivos titulares. En el caso de las distintas normas ISO mencionadas han sido desarrolladas por la International Organization for Standardization.

Se citan estas marcas porque se consideran referentes en los temas que se tratan, buscando únicamente fines puramente divulgativos. En ningún momento se busca con su mención el uso interesado de estas marcas ni manifestar cualquier participación y/o autoría de las mismas.

Nada de lo contenido en este documento debe ser entendido como concesión, por implicación o de otra forma, y cualquier licencia o derecho para las Marcas Registradas deben tener una autorización escrita de los terceros propietarios de la marca.

Por otro lado, se renuncia expresamente a asumir cualquier responsabilidad relacionada con la publicación de las Marcas Registradas en este documento en cuanto al uso de ninguna en particular y se eximen de la responsabilidad de la utilización de dichas Marcas por terceros. El carácter de este documento es únicamente conmemorativo del aniversario del comité y buscando en todo momento facilitar a los lectores la comprensión, adaptación y divulgación de las disciplinas, metodologías, estándares y normas presentes en el ámbito de la calidad del software.



Edición: Enero 2013

Diseño y maquetación: CSTIC
Asociación Española para la Calidad
Claudio Coello, 92
28006 Madrid
aec@aec.es
www.aec.es

“30 Años de Calidad no son Nada...” es una obra que recopila en parte fragmentos de artículos de diversos autores por lo que en cada artículo concreto se deberá contactar con cada autor para determinar las condiciones específicas de su utilización.

En su contexto general se encuentra bajo una Licencia Creative Commons



“Reconocimiento-NoComercial-SinObraDerivada”

Se permite su copia y distribución por cualquier medio siempre que mantenga el reconocimiento de sus autores, no haga uso comercial de las obras, ni la generación de obras derivadas...

Carta de la Dirección de la AEC



En nombre de la Asociación Española para la Calidad (AEC), le presento este libro conmemorativo del 30 aniversario del Comité de Calidad en los Sistemas y Tecnologías de la información y la Comunicación, en sus inicios Comité de Software de la AEC.



Año tras año, este comité ha ido evolucionando en función de las demandas de los profesionales y las organizaciones, siempre con el objetivo de ser un punto de encuentro de los especialistas en este área. De esta manera, siempre ha querido ser el foro ideal para la presentación y el debate de las últimas tendencias en su ámbito de actuación, así como ser el punto de encuentro para el intercambio de ideas y la exposición de sus avances metodológicos más prometedores. Con el intercambio de conocimientos, este comité tiene por objetivo contribuir a que las empresas españolas apliquen en mayor grado estas técnicas y métodos, con el fin de incrementar su competitividad mediante la mejora de la calidad de sus procesos, productos y servicios.

Este año, con la celebración de su 30 aniversario, al comité le ha parecido oportuno realizar un pequeño repaso de su actividad, sobre todo a través de los artículos publicados en la Revista "Calidad". Por ello, nos llena de orgullo apoyar al comité en dicha iniciativa y comprobar que se mantiene con paso firme a lo largo de tiempo.

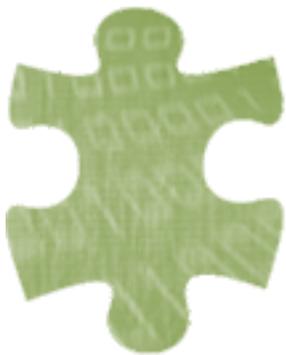
Deseo firmemente que este libro pueda ser de gran utilidad en su desempeño profesional.

Atentamente,

Marta Villanueva,
Directora General de la AEC



personas



procesos



tecnologías

Carta Presidente del Comité

Las personas, los procesos y las tecnologías. Estos tres elementos se han considerado tradicionalmente la base de las organizaciones modernas, capaces de mejorar sus procesos, realizados por personas, y optimizados con el uso de la tecnología. Y estos tres elementos han sido la preocupación constante de la Asociación Española para la Calidad (AEC).



Desde su fundación, hace ya más de medio siglo, la AEC estableció las bases para difundir, en nuestro país, las mejores prácticas que han permitido a innumerables organizaciones, conseguir ser más competitivas, apoyándose en sus procesos.

Y hace ya treinta años, esta Asociación decidió dar un paso más, creando el Comité Calidad del Software, con el fin de difundir las prácticas que pretendían asegurar la calidad de los productos software y sistemas de información que se desarrollaban en las organizaciones españolas. A día de hoy, éste Comité ha ampliado su alcance, a la vez que su denominación, para pasar a ser conocido como el Comité de Calidad de los Sistemas y Tecnologías de la Información y Comunicaciones (CSTIC).

Con motivo de este aniversario, se ha publicado esta colección de trabajos producidos en el seno del comité, donde el peso de la historia se hace evidente, representando el interés de nuestros vocales por el mundo de la calidad del software, pero sin olvidar por ello el resto de contenidos de nuestro alcance: la gestión de los servicios de tecnologías de la información, el buen gobierno de las TIC y la seguridad de la información.

Desde el puesto que me ha tocado desempeñar en estos días, aprovecho la oportunidad para agradecer la generosidad y la dedicación que han tenido todos los vocales que han formado parte de la historia de este comité, en sus distintas etapas, y felicitarles por las contribuciones aportadas.

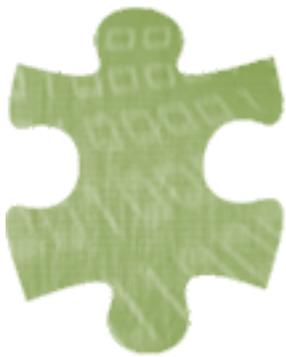
Animo a los lectores a considerar la oportunidad de pertenecer a este Comité, tanto si se trata de organizaciones pertenecientes al sector TIC, como si se trata de aquellos departamentos de tecnologías y sistemas de información de organizaciones pertenecientes a otros sectores, que se apoyan en las TIC para conseguir el cumplimiento de sus objetivos de negocio empresariales.

Espero que juntos podamos cumplir otros treinta años: muchas felicidades.

Ramiro Carballo
Presidente del Comité de Calidad de los Sistemas y
Tecnologías de la Información y Comunicaciones (CSTIC).
Asociación Española para la Calidad (AEC).



personas



procesos



tecnologías

Un poco de historia.

Hace 30 años, en 1982 se constituyó el Comité de Software tras detectarse la necesidad de mejorar la organización de los procesos informáticos en las grandes empresas de nuestro país, en concreto en la construcción de sistemas críticos de software y electrónica en el sector de la defensa.

A lo largo de estos años, la presidencia del Comité ha estado asumida en 1982, y como primer Presidente, por Gerardo Villanueva, de Standard Eléctrica. De 1983 a 2000, Julio González Sanz, de Telefónica de España. De 2003 a 2005, Francisco Ogalla, de Renfe. Y desde febrero de 2005 hasta la actualidad, Ramiro Carballo, de Caelum.

En 1983 se presentaba en la revista de la entonces AECC un artículo escrito por D. Julio González, en el cual se presentaba al nuevo Comité de Calidad del Software, el cual se había constituido unos meses antes.

El artículo comenzaba de la siguiente manera:

“La necesidad sentida por diversos sectores profesionales, industriales y comerciales de la A.E.C.C. de una mejor calidad tanto de los productos software como de los procesos que se llevan a cabo a lo largo de su ciclo de desarrollo para asegurar la calidad de aquellos, propició la constitución del Comité de Software.

Este artículo es una presentación de los objetivos, ámbito de aplicación, actividades, participantes y su presentación junto a la relación de los documentos generados y actividades realizadas hasta el momento presente por los miembros del Comité. Finalmente se muestran las actividades previstas a corto y medio plazo y la estrategia a largo plazo del Comité.”.

El comité quedó constituido por las siguientes personas:

Presidente: D. Julio González Sanz (C.T.N.E.)

/ TELEFONICA

Secretario: D. Pablo Corrales - MESA

Vocales:

D. Tomas Elices Concha - INTA

D. José Manuel García Conca - INTA

D. Luis Gil Alcázar- INTELSA

D. Joaquín Esteban - INTELSA

D. Ignacio Macho -SESA

D. F. de Diego - AMPER

D. Juan Pardo - CAMPM

D. Ignacio Pérez Bernat- ITS

D. Walter Strobl- ZTS

D. José Luis Prieto - EESA/ INISEL

Da. Clementina Bravo - INISEL

Da. Marta D'Amore- ERIA, S.A.

D. José Gómez González - J.E.N.

D. José María Granda - EMPRESARIOS

AGRUPADOS

D. Roberto Rodríguez Liñan- ABENGOA,

S.A.

El contenido completo del artículo lo tenemos al final de esta introducción ([enlace](#))

A partir de aquel primer artículo la presencia de los miembros del comité fue cada vez más común en las publicaciones de la asociación. Sus artículos eran potentes y curiosamente siguen manteniendo esa potencia a día de hoy.

Julio González comenzó escribiendo “Los productos software y su calidad hoy” ([enlace](#)) al que siguió otro de Marta D’Amore titulado “Aspectos metodológicos y normativos en el control de la calidad del software” ([enlace](#)). Tanto Julio que por entonces trabajaba en Telefónica como Marta que lo hacía en Eria eran los referentes que tanto los profesionales como los estudiantes de informática teníamos en aquellos años. Ellos empezaron a difundir tanto el uso de la calidad como de las metodologías en la informática de gestión de aquellos años.

Durante esa época los artículos que el comité publico fueron altamente reconocidos y tomaron una posición muy significativa dentro del panorama nacional; títulos como “Consideraciones sobre la influencia de la Organización en la Calidad Final del Software de un Sistema de comunicaciones”, “Un programa de Garantía de Calidad de Sistemas Complejos para la adquisición en Tiempo Real”, “La fase de los requerimientos en un Software Crítico” o “Estrategias y requisitos en la Prueba de Programas” fueron algunos de ellos.

I Congreso Ibérico de la Calidad en el Software y Sistemas de Información

En la incomparable ciudad de Lisboa y durante los primeros días 11 al 13 de octubre ha tenido lugar este I Congreso Ibérico de la Calidad en el Software y Sistemas de Información. El lugar, el Hotel Alfa Park inaugurado hace unos años, gracias a la compra de la EXPO 98, dotado con unas buenas instalaciones para este Congreso.

El origen del evento tuvo lugar durante la celebración en Lisboa del Congreso anual de la IEEE del año 1994 a la vista del interés mostrado por los participantes en las sesiones dedicadas a la calidad del software así como a la consolidación de los Congresos de Calidad del Software de la IEEE que en diciembre celebraban su 5º Edición en Lisboa. Desde la 1ª y 2ª edición en Lisboa, el Congreso portugués se pasó a la realización de un tema para tener en hora de debate las ideas y experiencias propias de estas tres empresas y organizaciones. Estas sesiones a participar tanto en el Comité Organizador como en el Comité Técnico del Congreso en representación de la AECQ por nuestra experiencia en todos los Congresos Europeos y particular por el Congreso de Madrid que presidimos en el año 1993, así como por la amistad y confianza de haber participado con ellos en otros congresos europeos.

A su vez, nuestra, se amplió el alcance del Congreso a fin de dar la oportunidad de la calidad de los Sistemas de Información, no sólo del software, sino de sistemas en sus innovaciones que todos los Congresos realizamos desde por la IEEE, como por la AECQ, el reciente congreso en San Francisco, con la participación de los dos organismos mencionados y la AECQ, se nos refirió a la Calidad del Software.

Entendemos que todavía queda bastante por hacer en dicha área, pero las causas principales de las deficiencias y errores ya una abundante literatura basada en la investigación y en la experiencia práctica. No aterra el hecho de que la calidad de los Sistemas de Información. La literatura y los libros sobre metodologías y aplicaciones en internet, por ejemplo respecto a los modelos, métodos, herramientas, técnicas etc., relativos a la calidad de los S.I. han alcanzado relevancia a nivel internacional. Parece evidente que la calidad de un S.I. proviene en parte, pero no sólo en parte, de la calidad del software.

La concepción de la calidad del software en un contexto europeo tiene su origen en las actividades de las instituciones técnicas de normalización y de la calidad de los sistemas, así como la calidad de los productos de información del software en las industrias, de los productos de información, de los sistemas que se ven afectadas, muchas veces de gran amplitud por numerosos análisis, modelos, normas, etc. que no sólo se aplica a sólo un producto sino a sistemas de información de información. Por tanto, una preocupación que nos lleva a pensar en la calidad del software, pero que también se aplica a otros sistemas de información que se ven afectados por la calidad del software.

El caso europeo, que se ha desarrollado para iniciar a nuestra Sección a nivel europeo, que quiere participar en el desarrollo de las ideas planteadas.

Estas consideraciones nos llevaron a pensar también al Comité de Software de la IEEE y algunos congresos que se celebraron en otros países que se celebraron con el tiempo y falta de tiempo de los congresos europeos.

El modelo de que nuestra actividad se refería sólo al mejoramiento de la calidad y responsabilidad entre otros para ampliar a las actividades de gestión, producción y mantenimiento del software.

La apertura estuvo presidida por el Presidente de la AECQ, Juan M. Ferrer (IIEP), el Vicepresidente del IQQ que mostraron un compromiso y apoyo que el título del Congreso se centra por el de las actividades de los otros organismos de trabajo. Después de las experiencias muy interesantes que se realizaron en los últimos días de una 148 jornada, con la presencia de más de 100 personas, con la asistencia de España y Portugal, los países de Francia, Italia, Alemania, Argentina y Suiza.

El Congreso tuvo la siguiente estructura: el primer día estuvo dedicado para iniciar sobre temas específicos. Una de las charlas a impartir por Alberto Parrondo, de INDEF, sobre los “Criterios y Metodología Europea de Evaluación de la Seguridad de los Sistemas de Información (el esquema ITSEC/ITSEM)”. La teoría que imparte se dedicó a la evaluación del “Estado actual de la práctica de las Normas de Ingeniería y Calidad del Software”.

Los días siguientes se organizaron en dos sesiones paralelas con tres sesiones especiales: apertura, cierre, una sesión especial de información y calidad de los sistemas de información y calidad de los sistemas de información en los Sistemas de Información y siete sesiones paralelas: Aplicación de ISO 9000 en el Software de Información; Los Procesos del Ciclo de Vida Software; la Gestión de la Calidad en el Desarrollo e Integración de los Sistemas de Información; los sistemas de Herramientas, Modelos y Métodos; y una sesión para la experiencia portuguesa y española del programa ITSEC/ITSEM.

En total 40 presentaciones que duraron un hora contada dentro del Congreso. Los participantes españoles estuvieron distribuidos en casi todas las sesiones y participaron en la mayoría de ellas. Entre los participantes españoles estuvieron: AMBA (AENA), José Ignacio Macho (INDEF), Carlos Gallego (INDEF), Juan M. Ferrer (IIEP), Andrés Palomares (INDEF), Isabel Gallego (AENA) y Julio González (AENA).

La calidad de la calidad que afecta a los Sistemas de Información así como el valor fundamental que los propios Sistemas de Información aportan a la calidad de los procesos de control, producción y de servicios, añadiendo un valor a los procesos del negocio que en muchos casos son el componente estratégico clave de diferenciación de los servicios prestados, instrumento técnico indispensable para conseguir una adecuada calidad y rendimiento competitivo en el mercado.

Por último nos queda agradecer el especial esfuerzo realizado por Isabel Alameda de la AECQ, de nuestro amigo de las Comisiones Técnica y Organizadora, Belén de Vicente (INDEF), Fernando Guerrero (IIEP), Fernando Garbajosa (AECQ), Adalberto Matilla (APQ) y Antonio Sima (Pico Waterhouse) y a todos los presentes españoles.

Julio González Ferrer
Presidente de la Sección de Software y Sistemas Informáticos

Ya a finales de los 80 y de la mano del entonces presidente del comité Julio González se publicó un artículo que planteaba por primera vez una situación que las empresas no veían todavía cómo solucionar: la formación.

El artículo se tituló: “La Formación: Una clave esencial en la Mejora de la Calidad del Software” ([enlace](#)) y tuvo una importante repercusión.

En el año 94 se celebró el I congreso Ibérico de la calidad en el software y Sistemas de información en Lisboa donde hubo una amplia representación de este comité ([enlace](#)).

Numerosas son las actividades llevadas a cabo por el Comité a lo largo de todos estos años y que podríamos clasificar en los siguientes apartados:

- Colaboración con organismos internacionales como IEEE, subcomité de Normas de Ingeniería del

Software del IEEE/CS de Estados Unidos, OTAN a través del Ministerio de Defensa, ISO en representación de AENOR en reuniones plenarios.

- Relaciones con organismos oficiales y de normalización nacionales (1987) con los CTN 66 “Gestión de Calidad” y CTN 73 “Industria Nuclear” en AENOR.
- Participación junto con asociaciones profesionales, ATI (Asociación de Técnicos Informáticos 2007), y con el Ministerio de Administraciones Públicas en trabajos relacionados con la seguridad de los sistemas informáticos.
- Participación y organización de seminarios, eventos nacionales e internacionales e iniciativas de calidad.
- Organización del Congreso “Oportunidades para la Industria Española del Software en la Economía Global”, con la colaboración del Ministerio de Industria, Turismo y Comercio (2006).
- Premio Calidad TIC, otorgado a Endesa en 2009, con accésit para INTECO y Vodafone.
- VII Congreso de Calidad en la Gestión de las TIC, CSTIC 2009.

A lo largo de sus 29 años de historia han sido editadas 17 publicaciones de gran utilidad práctica para el sector. Asimismo, ha colaborado con diversos artículos en la revista

CALIDAD.

En enero del año 2009, el Comité de Software amplió su alcance a todos los procesos organizacionales relacionados con las tecnologías de la información, pasando a denominarse Comité de Calidad en los Sistemas y Tecnologías de la Información y las Comunicaciones (CSTIC).

Llegando el nuevo milenio el comité fue creciendo, hubo nuevos presidentes, más empresas que se añadieron al mismo y como nota uniforme los artículos siguieron apareciendo con interés por ello hemos estructurado los mismos de la siguiente manera:

La calidad del software en el siglo XXI

La aparición de Internet, redes sociales, nuevos dispositivos e innumerables conexiones e integraciones entre música, imagen, sonido, texto y gráfico hacen de las Tecnologías de la Información un nuevo ecosistema donde muchos paradigmas han tenido que ser definidos, otros ha habido que volver a inventarlos y alguno que otro sigue vigente.

La calidad del software no ha sido ajena a esta evolución; las actividades básicas de estimación, programación, planificación, inspección, etc., siguen vigentes pero la estrecha unión que nace con la tecnología y los procesos dan nuevas dimensiones a conceptos como la mejora, Lean, Gobierno Corporativo, seguridad, Cloud, los cuales obligan a redefinir nuevos puntos de vistas y conceptos que teníamos claros hasta el momento

De hecho el mismo comité de software se redefine como Comité de Calidad de los Sistemas y Tecnologías de la Información y Telecomunicaciones (CSTIC), lo cual le dota de un ámbito mucho más amplio que el ofrecido hasta ahora.

Por ello hemos agrupado los artículos más interesantes publicados en apartados que muestren esta relevancia.

Artículos presentados en esta selección:

- [Presentación del Comité de Calidad de Software de la A.E.C.C.](#)
- [Los productos Software y su Calidad hoy](#)
- [Aspectos metodológicos y normativos en el Control de Calidad del Software](#)
- [La Formación: Una clave esencial en la Mejora de la Calidad del Software](#)
- [I Congreso Ibérico de la Calidad en el Software y Sistemas de Información](#)
- [Las empresas de Tecnologías de la información y la nueva economía](#)



personas



procesos



tecnologías

Presentación del Comité de Calidad de Software de la A.E.C.C.



La necesidad sentida por diversos sectores profesionales, industriales y comerciales de la A.E.C.C. de una mejor calidad tanto de los productos software como de los procesos que se llevan a cabo a lo largo de su ciclo de desarrollo para asegurar la calidad de aquellos, propició la constitución del Comité de Software.

Este artículo es una presentación de los objetivos, ámbito de aplicación, actividades, participantes y su presentación junto a la relación de los documentos generados y actividades realizadas hasta el momento presente por los miembros del Comité. Finalmente se muestran las actividades previstas a corto y medio plazo y la estrategia a largo plazo del Comité.

1. INTRODUCCIÓN

En un mundo industrial altamente tecnificado y competitivo como es el de los sistemas (de información, de comunicación, de armas, aeroespaciales, etc.) en los que el software participa controlando, evaluando, ordenando, etc. y asumiendo cada vez más funciones e importancia económica hasta el punto de que tiende, en los próximos años, a ocupar hasta el 90 por ciento del coste total del sistema, resulta que la calidad de tales productos un componente esencial siendo el momento en que los usuarios y la industria comienzan a sensibilizarse acerca de la importancia crítica que tiene respecto de sus intereses.

La A.E.C.C., al igual que otras Asociaciones empresariales y profesionales europeas y americanas tales como la IEEE/CS (U.S.A.), la A.F. C.I.Q. (Francia), etc., sintió la necesidad de crear un comité para el estudio y análisis de las características y problemática de los productos software desde el punto de vista de la Calidad.

El objetivo del comité es el de proporcionar a la industria y a los usuarios orientación, información y formación a través de guías, recomendaciones, seminarios y el propio debate interno del Comité. El comité de software está agrupando de una parte, a un número creciente de profesionales de diseño y desarrollo software que se enfrentan a problemas de calidad del producto diseñado, de otro con profesionales de aseguramiento y control de calidad que tienen la necesidad de revisar e inspeccionar cada día más sistemas que incorporan un producto abstracto, complejo, caro que controla dichos sistemas y que se llama software. Además están

los usuarios de dicho software que expresan sus requerimientos de calidad cada vez de forma más precisa y estricta.

El objetivo de esta presentación es la de exponer los objetivos generales del comité, su ámbito de aplicación, funciones, participantes y representación actuales, relaciones con otros organismos que tratan temas análogos, las actividades en curso y las previstas a medio y largo plazo. En un apéndice se da la lista de documentación de trabajo generada por los miembros del Comité.

2. OBJETIVOS

2.1. General

Desarrollar y difundir **métodos, técnicas y procedimientos para especificar, diseñar, construir, poner a punto, adquirir, aceptar y evaluar un producto software**, con el fin de garantizar un funcionamiento correcto del mismo en relación a los requerimientos y especificaciones del mismo teniendo en cuenta las limitaciones de recursos (dinero, personal, equipos y tiempo) y el conjunto de componentes de calidad exigido.

2.2. Otros Objetivos

- a) Orientar las técnicas y procedimientos para que disminuyan los costes de garantía de calidad ocasionados por: la detección y corrección de errores, modificaciones, optimizaciones, simulaciones, controles, etc...
- b) Normalizar formularios, técnicas y procedimientos de medida de los parámetros de calidad del software y el establecimiento de índices globales y parciales definitorios de la misma
- c) Establecer un glosario con definiciones de términos específicos y afines de calidad del software.

3. AMBITO DE APLICACION

Todos los aspectos relacionados con la calidad del software, tanto de gestión como de aplicación industrial y científica.

Existen muchas facetas relativas a la Calidad del Software (CS) que agrupamos en dos categorías según pertenezcan al producto acabado o al proceso de diseño y desarrollo.

A la calidad del producto se refieren facetas tales como

- selección del proyecto
- los requerimientos y especificaciones del producto (rendimientos y funciones del mismo)
- los requerimientos de instalación
- mantenimiento
- operación y explotación
- etc.

A la calidad del proceso se refieren los:

Aspectos técnicos	
Las metodologías y normas de diseño y desarrollo de	Metodología y normas para
• Especificación de requerimientos	• Modelización
• Análisis del Sistema	• Recogida de Datos
• Diseño de Detalle	• Medidas de calidad
• Programación y codificación	• Medidas de productividad
• Puesta a punto e integración	• etc.
• Pruebas de aceptación del sistema	
• etc.	
Aspectos de la Gestión del software	
- Organización y planificación de la función de garantía de calidad.	
- Técnicas formales de revisión e inspección.	
- Gestión de configuraciones.	
- Control de cambios y de versiones.	
- Control de proyectos (tiempo y recursos).	
- Productividad.	
- Costes.	

El análisis de las relaciones entre calidad de producto y calidad del proceso resulta de gran interés puesto que a veces sucede que el esfuerzo en optimizar el proceso productivo no es suficiente para obtener un producto de calidad si los requerimientos son inadecuados, si las especificaciones del usuario son incorrectas o si la elección del proyecto por parte del Comité de selección de proyectos ha sido inadecuada.

De otra parte es posible obtener un producto de calidad con un proceso de producción de baja calidad y eficiencia, tal como ocurre en la producción de objetos de artesanía realizados en plantas poco eficientes con pobres recursos físicos y de organización y gestión de la producción.

Por ello, una práctica frecuente para la obtención de un producto de calidad es la implantación de un sistema de control de calidad centrado en descubrir errores y corregirlos antes de que el producto llegue al usuario. Esto es un método caro por lo que resulta más beneficioso dedicar el dinero a la prevención de errores mejorando el proceso global de producción, y los métodos particulares de realización, así como la formación del personal con el objetivo general de alcanzar una mayor calidad paralelamente a un aumento de productividad. Ello lleva consigo la necesidad de implantar un sistema para la Gestión de la calidad del software.

4. FUNCIONES

1. Presentación y análisis de los problemas más frecuentes respecto a la calidad del software (C.S.)
2. Desarrollo de técnicas para el control de la C.S.
 - 2.1. Determinación de las características deseables de un producto software (por ejemplo: fiabilidad, portabilidad, modularidad, etc.)
 - 2.2. Determinación de parámetros y técnicas de medida para cada una de las

características deseables.

- 2.3. Determinación de expresiones, que en función de los parámetros de índices o indicadores de la C.S.
3. Análisis del impacto de la C.S. en su mantenimiento y adaptación. Reducción de los costes.
4. Desarrollo de textos para formación en el control y gestión de calidad del software. Planificación de cursos y seminarios, etc.
5. Organización de charlas, conferencias, etc. sobre los diversos aspectos del tema.
6. Normalización de procedimientos y métodos, formularios, términos, índices, etc.
7. Intercambios de experiencias entre los miembros.
8. Preparación de informes y trabajos sobre temas y experiencias específicos sobre calidad del software. Publicación y difusión.
9. Mantener contactos con organizaciones similares.

5. PARTICIPACION Y REPRESENTACION ACTUAL

El Comité de C.S. pertenece a la Asociación Española de Control de Calidad (A.E.C.C.) y ha iniciado sus actividades hace prácticamente un año.

Su organización está abierta a todos los organismos y empresas públicos y privados nacionales o multinacionales con representación en el Estado Español, que pertenezcan por voluntad propia a la A.E.C.C. y que deseen participar en sus actividades.

A medida que el comité se va dando a conocer a través de contactos personales, jornadas, coloquios, seminarios, conferencias, se van incorporando nuevos componentes al mismo y en la actualidad pertenecen al Comité las siguientes personas, soportadas por sus respectivas empresas u organismos:

Presidente: D. Julio González Sanz (C.T.N.E.) / TELEFONICA

Secretario: D. Pablo Corrales - MESA

Vocales:

D. Tomas Elices Concha - INTA

D. José Manuel García Conca - INTA

D. Luis Gil Alcázar- INTELSA

D. Joaquín Esteban - INTELSA

D. Ignacio Macho -SESA

D. F. de Diego - AMPER

D. Juan Pardo - CAMPM

D. Ignacio Pérez Bernat- ITS

D. Walter Strobl- ZTS

D. José Luis Prieto - EESA / INISEL

Da. Clementina Bravo - INISEL

Da. Marta D'Amore- ERIA, S.A.

D. José Gómez González- J.E.N.

D. José María Granda -EMPRESARIOS
AGRUPADOS

D. Roberto Rodríguez Liñán- ABENGOA,
S.A.

De su composición actual vemos que no existe representación de centros oficiales, universidades, ningún constructor tradicional de la industria informática de gestión, siendo mayoritaria la representación de la industria electrónica y comunicaciones, tendiendo a aumentar la representación de las sociedades de servicio.

Esta falta de representación de importantes sectores la achacamos a consecuencia de:

a) Poco tiempo de funcionamiento del Comité y por ello poca difusión del mismo y desconocimiento del posible ámbito de actividades y de sus funciones.

b) Las propias empresas no se suelen comprometer en actividades que estén dirigidas a la realización de recomendaciones o normativas de tipo general que no coincidan exactamente

con las suyas.

Todo esto representa un desafío importante al Comité:

- a corto plazo, porque ha de existir mayor representación para que tengan crédito sus actividades.
- a medio plazo, porque es preciso elaborar recomendaciones de alcance nacional e internacional que ayuden tanto a los usuarios como a los constructores en sus desarrollos, evaluaciones, contratos de compra de servicios y productos, etc.

6. RELACIONES CON OTROS ORGANISMOS

En el momento actual el Comité de Software tiene relaciones directas con el Comité de Software de la EOQC con el que ha tenido reuniones en Madrid, Brihgton, Toulouse y Lisboa.

De otra parte algunos de sus componentes participan en otros organismos internacionales como la Comisión X del CCITT, para Lenguajes y Métodos de especificación de aplicación en Telecomunicaciones, con cuestiones de estudio específicas del aseguramiento de la calidad y de la fiabilidad del Software. A su vez dicha Comisión tiene conexión con el TC56 de la CEI con el grupo de trabajo WG10 cuyas actividades están dirigidas al estudio de diversos aspectos sobre la fiabilidad y mantenibilidad del software.

Se tiene participación activa en varios proyectos de normas de la IEEE/Computer Society relacionada con diversos temas de ingeniería y calidad del Software.

Existe conexión con la Comisión C del Subcomité de Ingeniería Software, de la Asociación Hispanoamericana de Centros de Investigación y Estudios de Telecomunicaciones que agrupa a 19 países de habla hispana.

El Comité está abierto a prestar su 'colaboración con otras asociaciones empresariales Y profesionales cuyos objetivos puedan coincidir en parte con los suyos para evitar duplicaciones de esfuerzos y armonizar sus respectivas orientaciones, documentaciones creadas, etc. De especial interés pueda ser la colaboración con la nueva AENOR (Asociación Española de Normalización) a la que desde aquí brindamos la experiencia adquirida durante los tres años de funcionamiento de aquel.

7. ACTIVIDADES

Las actividades del Comité son de diversa índole. Al menos hay una reunión del mismo cada mes y en el que se debate, se intercambia información, se orientan los trabajos a realizar y se revisan, etc. En el Apéndice 1 se da la lista de documentación interna generada por diversos miembros del Comité.

De relevancia especial han sido la Jornada - Coloquio celebrada el pasado día 22 de octubre sobre la "Calidad de los Productos Software y su importancia hoy en las relaciones clientes-suministradores".

Se ha realizado el 1 Seminario sobre Calidad del Software el pasado 14 y 15 de noviembre patrocinado por la A.E.C.C. y el IRANOR, y se ha colaborado en el Seminario sobre la "Gestión y Control de Calidad en la Industria Electrónica" patrocinado por la Fundación General de la Universidad Politécnica de Madrid dentro del Programa de Tecnología de Calidad para Posgraduados.

Las actividades a corto plazo son las siguientes:

- a) Publicación de un Glosario de Términos de Ingeniería y Calidad del Software.
- b) Publicación de una recomendación y guía para los planes de aseguramiento de la Calidad del Software.

Las actividades a medio plazo son:

- a) Preparación de una jornada coloquio sobre calidad del Software a celebrar durante el III Congreso Nacional de Calidad en Zaragoza.
- b) Preparación contribuciones al Comité de Software de la EOQC con ocasión de la 30 Conferencia anual a celebrar en Estocolmo.
- c) Elaboración de guías y recomendaciones sobre:
 - Los planes de gestión de configuraciones del Software.
 - Los planes de verificación y validación del Software.
 - Aspectos contractuales respecto la calidad del Software en las relaciones cliente/proveedor.
 - Guía para la Documentación de Sistemas Software.
 - Guía de útiles para el aseguramiento de la calidad del Software.
 - etc.

Algunos de estos proyectos pudiera requerir financiación externa a la A.E.C.C. por lo que habrá que contactar y establecer los cauces adecuados a la misma.

A largo plazo (más de 2 años), la estrategia de desarrollo del Comité estará dirigida a:

- a) Consolidar el Comité en cuanto a su reconocimiento por los profesionales, industria, usuarios y organismos afectados por su actividad.
- b) Ayudar a los centros para la promoción en las autonomías en cuanto a la calidad del Software se refiere (estos centros son ya hoy los de Aragón, Asturias, Cataluña, Navarra, País Vasco y Valencia).
- c) Crear grupos de trabajo sobre temas específicos:
 - Fiabilidad del Software,
 - Métricas de calidad,
 - Análisis de Costes/Calidad,
 - etc.
- d) Participación activa en comisiones de Normalización a nivel nacional e internacional.

Los productos Software y su Calidad hoy

Por Julio González Sanz Ingeniero de Montes y Licenciado en Informática TELEFONICA, Dept. Investigación y Desarrollo Presidente del Comité de Calidad del Software de la AECC.



INTRODUCCION

El objetivo de esta comunicación es el de presentar una panorámica de los conceptos en torno a los términos "Calidad del Software", refiriéndonos a la problemática general de los productos software y al impacto de la calidad en los mismos y, como consecuencia, a la necesidad de implantar la Función de Garantía de Calidad del Software.

El concepto de calidad está configurado por factores tales como la fiabilidad, la mantenibilidad, la eficiencia, la portabilidad, etc. Con objeto de poder cuantificar estos factores de calidad se dividen en criterios que representan los atributos de calidad desde el punto de vista del propio producto Software o de su proceso de producción, y por los que se puede definir o medir la calidad. Se establecen métricas de calidad para proporcionar una medida cuantitativa de los atributos representados por los criterios.

La garantía de que un producto funciona de acuerdo a unas especificaciones, tradicionalmente está proporcionada por unas pruebas del producto al final de su ciclo de desarrollo. En el caso del Software, y en especial cuando es de gran escala y considerado crítico, estas pruebas finales no se muestran suficientes.

El término "crítico" se expresa en el sentido de que un fallo puede afectar a la seguridad física de las personas o causar importantes pérdidas monetarias.

Para ello es necesaria una metodología que permita asegurar que el Software adquiere las características requeridas a medida que progresan. Dicha metodología ha tenido varias expresiones normalizadas dentro del contexto militar U.S.A., sin embargo la industria civil ha de disponer de normas que expresen el método de lo que debe contener un plan de Garantía de Calidad para el Software (G.C.S.). Tanto en U.S.A. como en Europa se han tomado acciones al respecto. En España se han sentido las mismas necesidades y a través del Comité de Calidad hay iniciadas varias actividades orientadas en la misma dirección.

DATOS ECONOMICOS DEL SOFTWARE, HOY

Con objeto de mostrar algunos datos económicos de interés se presentan aquí dos cuadros elaborados por consultores internacionales. En el cuadro núm. 1 se dan las cifras de la evaluación del mercado de Software y Servicios en España en 1984. El valor total asciende a 58.757 millones de pesetas, de las que 14.847 millones corresponderían al concepto de Software empaquetado, lo que representa un 31% del total.

En el cuadro núm. 2 se dan las cifras del gasto informático en España para el período 1983-89. Las cifras aportadas para software y servicios indican un incremento en términos globales de un 400% para dicho período.

La magnitud de estas cifras pone en evidencia los importantes intercambios comerciales que van a presenciar a medio plazo los productos software.

ESPAÑA MERCADO DE SOFTWARE Y SERVICIOS. AÑO 1984			
	Sm	Ptas M	% Participación
FABRICANTES DE HARDWARE			
Paquetes de software	90	14.449	25 %
Software a medida/Asesoría	28	4.495	8 %
	118	18.944	32 %
CASAS DE SISTEMAS			
Software empaquetado	16	2.569	4 %
Software a medida/Asesoría	20	3.211	5 %
	36	5.780	10 %
VENDEDORES INDEPENDIENTES			
Paquetes de software	37	5.940	10 %
Software a medida/Asesoría	48	7.706	13 %
	85	13.646	23 %
Formación	19	3.050	5 %
Facilidades de gestión	2	321	1 %
SERVICIOS DE PROCESO			
Batch local	69	11.077	19 %
Resolución problemas remotos ..	14	2.248	4 %
Transacción remota	23	3.692	6 %
Subtotal	106	17.017	29 %
TOTAL	366	58.757	100 %
Tipo de cambio = 160,54 pesetas por dólar			
Fuente: IDC International			

El fuerte tirón del consumo está llamado a la participación en este mercado de inversiones que, aunque tímidas aún, no dejan de crecer y en las que es fácil intuir una gran competencia para repartirse el pastel teniendo en cuenta además de la penetración especialmente de los productos "made in USA" que cubren cada día más estrechamente los requerimientos de los usuarios, esto es, proporcionando calidad a sus productos y a precios increíblemente bajos en muchas ocasiones.

Sin duda la producción nacional se habrá de dirigir en general a proyectos y

aplicaciones "a la medida" ya que la competencia en productos estándar para sistemas es, mal que nos pese, extremadamente dura... Si a ello añadimos los productos que nos presenten los países de la CEE el panorama aún se pondrá peor.

En la literatura técnica general existen escasos

datos relativos a los costes de Sistemas. Presentamos aquí la figura 1 tomada de un estudio del grupo DIEBOLD sobre tendencias e innovaciones en tecnología de Sistemas de Información que tendrán impacto durante los próximos tres a cinco años.

En dicha figura se muestran las extraordinarias reducciones de los costes previstos para el hardware. Sólo los costes de personal y del desarrollo e implantación de sistema software siguen aumentando a unos niveles considerables.

ASPECTOS GENERALES DEL SOFTWARE, HOY

La expansión de los ordenadores o computadores en los últimos años se debe a dos factores esenciales:

- Disminución de los costes del equipo físico
- Integración de los Sistemas de Computadores y Comunicaciones (C+C)

Que han propiciado la evolución del Proceso de Datos de los años 60, a los Sistemas de Información de los 70, pasando al Proceso Avanzado de Información de los actuales 80, para evolucionar en los próximos 90 a la llamada "Quinta Generación de Máquinas", con capacidad de utilizar procesos de tipo humano.

La situación actual y las tendencias previsibles incluyen la demanda de nuevo Software, estimándose que se multiplicará por 10 en la próxima década. Este Software es de todo tipo: de

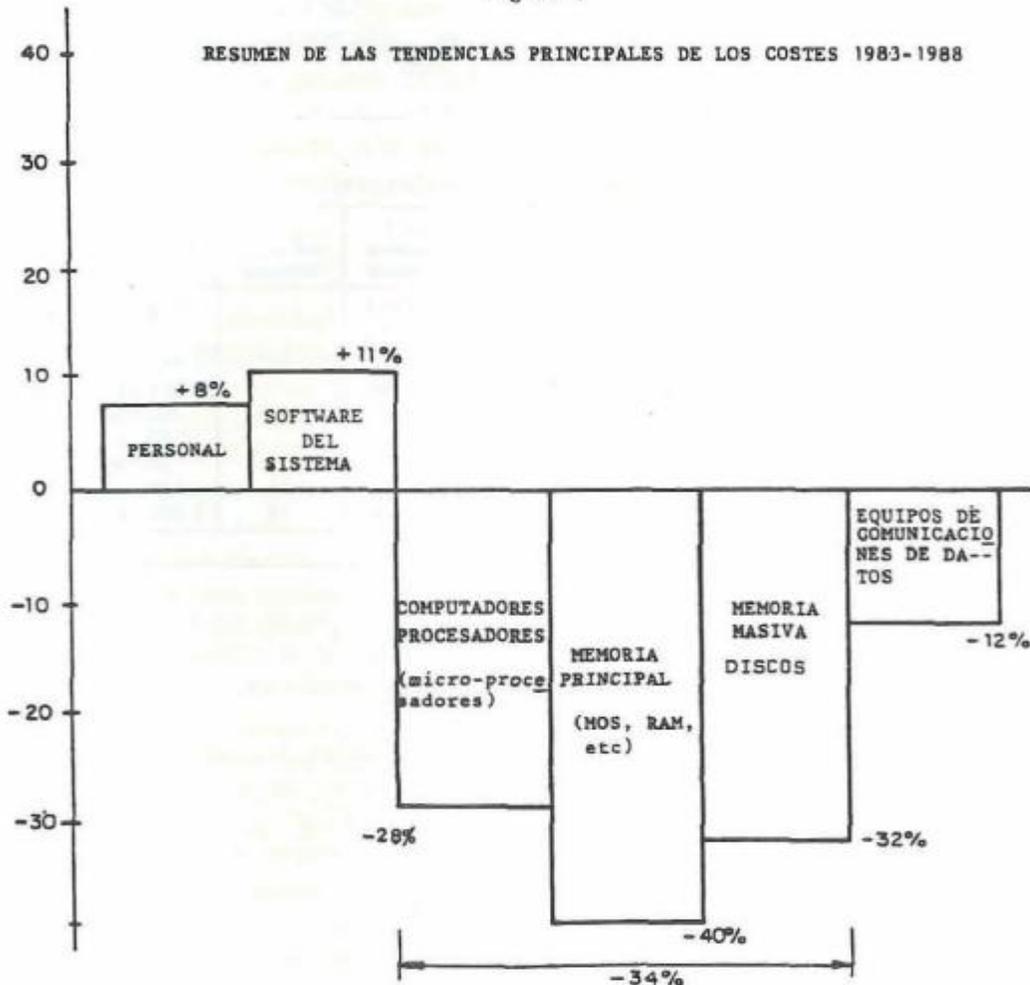
Gasto informático en España (1983-1989)

	1983 (Millones de pesetas)	Crecimiento medio anual (Porcentaje)	1989 (Millones de pesetas)
Personal	163.755		378.780
Gastos generales	17.250		35.910
Total gasto interno	181.005	15	414.690
Hardware	120.090		321.135
Software y servicios	49.448		196.500
Suministros	20.535		67.710
Comunicaciones	9.390		39.405
Total gasto externo	199.463	21	624.750
Total gasto informático	380.468	18	1.039.440

Fuente: 1984, Industry Briefing Session, IDC.

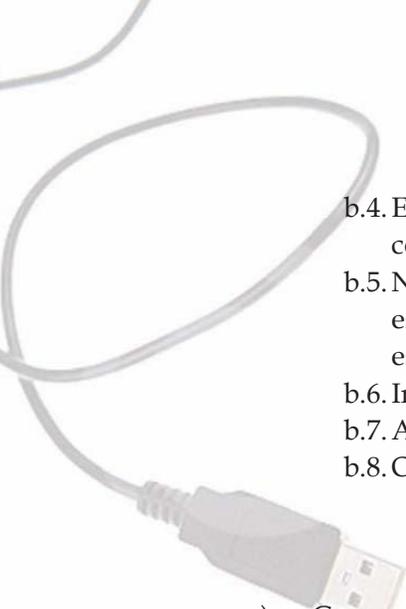
comunicaciones, de mando y control, de aplicaciones domésticas, etc., acentuándose cada día más las exigencias en la calidad de dichos productos.

Figura 1



Apuntamos aquí tres aspectos de los productos Software en relación a su incidencia en los aspectos de calidad de los mismos: naturales, problemática general, causas de dichos problemas.

- a) La naturaleza de los productos software
 - a.1. El software es un producto lógico y no físico hecho que significa que los costes están concentrados en su diseño y desarrollo y no en su proceso de producción.
 - a.2. Los productos software suelen realizar mayor variedad de funciones que los productos de Ingeniería normal.
 - a.3. Un único producto software puede implantarse en diversos entornos de explotación. Por ejemplo, un compilador portable
 - a.4. Los productos Software, construidos con metodologías adecuadas, permiten su evolución.
- b) Problema general Los problemas generales, sin excluir otros posibles, son:
 - b.1. Aumento constante en el tamaño y complejidad.
 - b.2. Carácter dinámico e iterativo a lo largo de su ciclo de vida.
 - b.3. Dificultades en conseguir productos totalmente depurados, causa de baja calidad.

- 
- b.4. En consecuencia, hay que dedicar altos recursos financieros a su mantenimiento correctivo.
 - b.5. No suelen estar terminados ni en los plazos previstos, ni con los costes estipulados, ni cumpliendo en el nivel deseable con los requerimientos especificados por el usuario (Baja calidad).
 - b.6. Incrementos constantes en los Costes de Desarrollo por baja productividad.
 - b.7. Alta dependencia de los proveedores.
 - b.8. Otros problemas propios de tecnologías inmaduras:
 - i. Procedimientos artesanales de producción software
 - ii. Falta de procedimientos normalizados de evaluación de calidad
 - iii. etc.
 - c) Causas de dichos problemas Las causas son de carácter diverso. Se apuntan aquí las que consideramos más relevantes:
 - c.1. Ausencia de unas especificaciones completas, coherentes y precisas del producto al momento de iniciar su diseño.
 - c.2. Ausencia, en la aplicación de métodos, de procedimientos y normas de diseño y desarrollo según una disciplina de Ingeniería Software.
 - c.3. Ausencia de la aplicación de las técnicas actuales de gestión de proyectos Software.
 - c.4. Escasez de personal con buena formación y experiencia en Software básico, de comunicaciones, microprogramación, etc...
 - c.5. Escasez de entornos de programación automatizados e integrados, incluyendo el desconocimiento de los disponibles en el mercado y de sus posibilidades de aplicación.
 - c.6. Otras, que dependen de las propias exigencias de cada compañía y organismo en particular, así como de su grado de desarrollo técnico y organizativo.

Nuestro punto de vista es, que para conseguir los objetivos de productividad y calidad exigibles a proyectos Software, hay que disponer de las normas de Ingeniería Software necesarias, y cuya columna vertebral está en una norma para los planes de Garantía de Calidad del Software, la descripción de cuyas funciones y actividades va a indicar el conjunto de normas mínimas de las que hay que disponer para la implantación sistemática de dichos planes.

Los beneficios principales aportados por la normalización del proceso de producción y productos Software, se derivan de la simplificación y facilidades a la hora de comparación de ofertas, transferencias de tecnología mantenimiento y explotación de Sistemas, formación del personal, lo que consecuentemente redundará en una mejora sustancial de la calidad de los productos y servicios.

DEFINICIONES Y CONCEPTOS DE CALIDAD Y GARANTÍA DE CALIDAD DEL SOFTWARE, HOY

La experiencia ha demostrado que la calidad del software es responsabilidad de todos los que participan en su realización o en su uso. La calidad debe ser "construida" en cada etapa de su ciclo de desarrollo. No se puede "añadir" calidad en fechas posteriores y por personal ajeno a dicho desarrollo.

A continuación se indican un conjunto de definiciones normalizadas que son la base de los análisis y descripciones posteriores.

Hay diferentes definiciones de los conceptos de calidad, garantía de calidad del software, etc. Se han seleccionado aquí aquellas que están respaldadas por organismos acreditados de normalización.

Calidad: es la totalidad de características de un producto o servicio relativas a su capacidad para satisfacer unas necesidades dadas (Norma ANSI/ASQC A3-1978).

Garantía de Calidad: es un modelo planificado y sistemático de todas las acciones necesarias para proporcionar una confianza o seguridad adecuada de que el elemento o producto está conforme a los requerimientos técnicos establecidos (MIL-STD-109B/DOD-D4 155.11/ANSI-IEEE STD 730-1984. Rev. 1).

También son aceptadas las siguientes definiciones:

"Todas aquellas acciones planificadas o sistemáticas necesarias para proporcionar una confianza adecuada de que un producto o servicio satisfaga necesidades dadas". (ANSI/ASQC A31978).

(2) El grado al que llega el software para una combinación deseada de atributos (por ejemplo fiabilidad, portabilidad y operatividad).

(3) El grado con el que un cliente o usuario percibe que el software cumple su o sus expectativas compuestas.

La calidad del software está compuesta por un conjunto de factores de calidad que son condiciones o características que contribuyen de manera activa a la calidad del producto software y todas ellas orientadas desde el punto de vista de la dirección o responsable de la adquisición e implantación del producto.

Para introducir una dimensión de cuantificación a los factores de calidad se definen un conjunto de criterios para cada factor.

Los criterios de calidad se refieren a aquellos atributos del software o del proceso de producción software que son independientes y a través de los cuales se puede juzgar, definir y medir la calidad.

Algunos criterios además de definir el factor de calidad asociado ayudan a describir las relaciones entre factores ya que varios de aquellos determinan más de un factor.

Así por ejemplo el criterio de Modularidad, definido como aquellos atributos del software que proporcionan una estructura del mismo de módulos altamente independientes, está relacionado con los factores de mantenibilidad, flexibilidad, facilidad para ser probado, portabilidad, reusabilidad e interoperabilidad.

Control de calidad son aquellas acciones de Garantía de Calidad que proporcionan un medio a controlar y medir las características de un elemento, proceso o facilidad respecto a los requerimientos establecidos.

El control de calidad representa una evaluación del producto o servicio a través de todo el ciclo de vida: especificación, diseño, producción o instalación, inspección y utilización.

Una definición alternativa de control de calidad es: "Las técnicas operativas y las actividades que sustentan la calidad de un producto o servicio que satisfará unas necesidades dadas: también el uso de tales técnicas y actividades" (ANSI/ASQC A3-1978).

La evaluación cuantitativa de los criterios se realiza a través de una o más métricas. Una métrica de Calidad es una medida cuantitativa del grado que tiene el software de un atributo dado y que afecta a su calidad (ANSI/IEEE Std 729-1983).

La Figura 2 muestra la estructura de la Calidad del Software según J.A. McCALL.

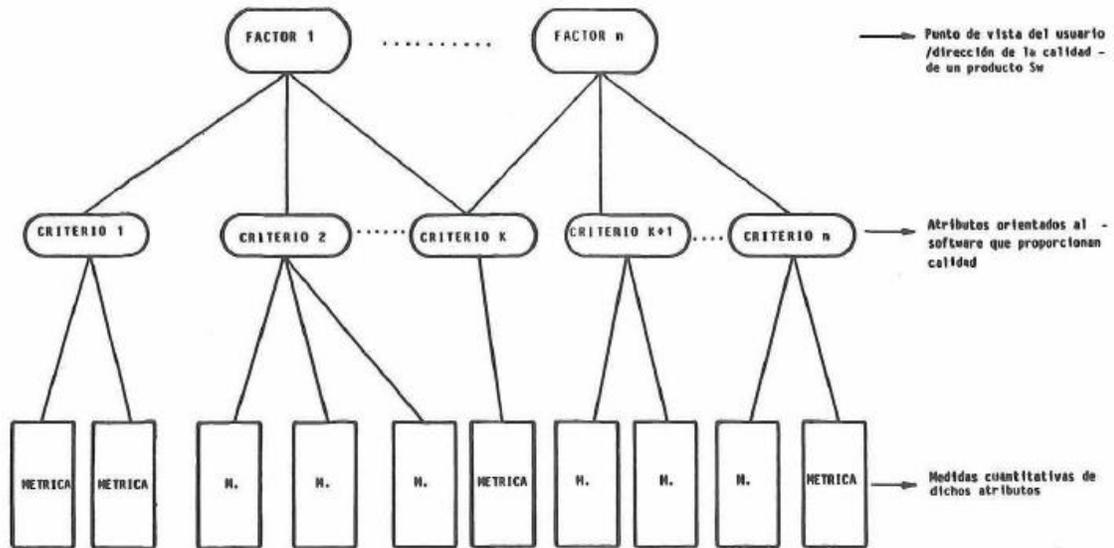


Figura 2

ESTRUCTURA DE LA CALIDAD DEL SOFTWARE (Según J.A. Mc CALL)

LAS CARACTERÍSTICAS DE LA CALIDAD DEL SOFTWARE

Los intentos más importantes de definición de características de Calidad del software son los realizados de un lado por McCall, Richards y Walters y de otro por Boehm, Brown, et al. Ambos son similares pero difieren en algunas de las construcciones y métricas propuestas.

Por ejemplo para el término autodestructiva para Boehm et al., está relacionado con los términos intermedios de comprensibilidad y capacidad a ser probado ambos sirven para medir la utilización primaria de mantenibilidad. Para McCall et al., autodestructivo está relacionado con varios factores pertenecientes al dominio de revisión y transición.

De otra parte, hay que tener en cuenta las interrelaciones de cada uno de los factores con los demás.

Los dos intentos se han desarrollado a partir de un conjunto jerarquizado de características del software.

La construcción de nivel superior en cada grupo representa

1. el comportamiento actual del software (explotación del producto)
2. la facilidad de cambio del mismo (revisión del producto)
3. la facilidad de conversión del sistema (transición del producto).

En la Figura 3 se muestran los dos modelos de calidad del software.

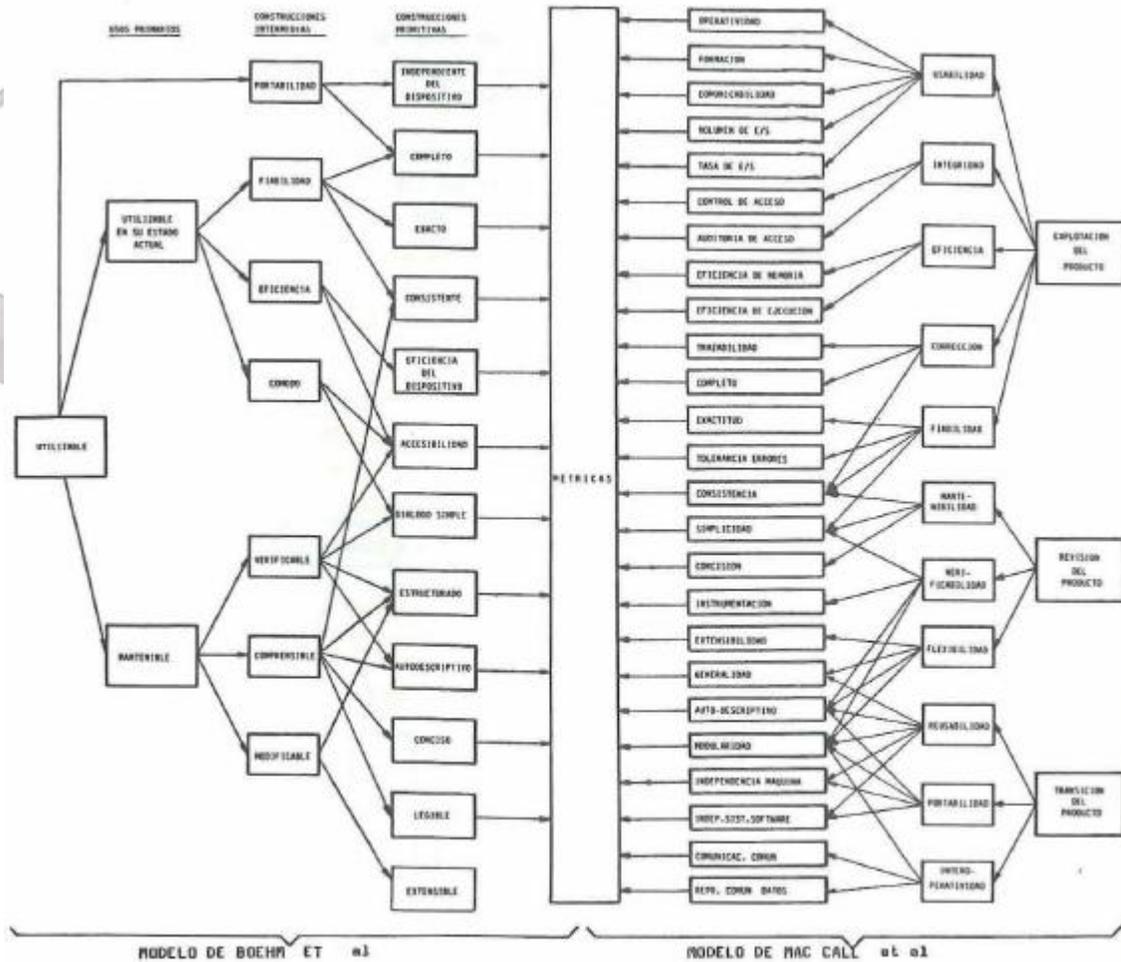


Figura 3 MODELOS DE CALIDAD DEL SOFTWARE

Ningún proyecto software puede, dentro de un presupuesto razonable, maximizar todos los factores de calidad. La naturaleza del sistema en desarrollo determinará la ponderación adecuada de los factores de calidad que se requieran como prioritarios, sin embargo hay que tener en cuenta la propia dinámica del software.

Así en sistemas tales como los de tiempo real en que la eficiencia en memoria y tiempo de respuesta puede ser un factor determinante, la optimización del código frecuentemente disminuye esta prioridad dejando paso a factores tales como la mantenibilidad.

La Figura 4 muestra dichas interrelaciones según McCall et al. que indican que existen algunos factores de calidad que son contradictorios y que los factores o características críticos, deben definirse desde el punto de vista del proyecto o sistema a desarrollar. Cada sistema software debe evaluarse en función de dichas características.

IMPLANTACION DE LA FUNCION DE GARANTIA DE CALIDAD DEL SOFTWARE (GCSW)

La implantación de la función de GCSW está determinada por la definición de los objetivos y principios generales que constituyen por sí mismos la Política de GCSW de la empresa, que en un documento expresa que se espera obtener de dicha función y las reglas y principios que

guían dicha implantación. Esta política puede estar inspirada o no en una norma externa (p. ej, MIL-STD-52779A), que ayuda en dicha definición de requerimientos de GCSW.

En otros términos puede expresarse la política de GCSW como las acciones orientadas a la verificación de cada actividad definida a frente a los requerimientos del sistema e insistiendo en la adecuada planificación a lo largo del ciclo de vida con objeto de minimizar problemas en las fases finales del ciclo.

Partiendo de la política de GCSW, se definen los métodos y medios generales para la realización de la GCSW, determinado en los tres documentos siguientes:

1. La Norma para los Planes de GCSW (NPGCSW) en la que estarán contenidas cuales son las actividades a realizar. Esta Norma es un esquema general que proporciona la estructura general de un PGCSW específico, esto es, da las cabeceras de los capítulos que deben aparecer en el Plan específico.

2. El Manual de Calidad Software (MCSW) en el que se describen los procedimientos, métodos, técnicas, útiles y normas a utilizar para construir la calidad del producto durante su desarrollo y para controlar la misma.

3. La Guía de Aplicación del PGSW es un Documento en que sigue la estructura de la Norma explicando el contenido de cada capítulo e indicando en cada caso el útil, norma, método, etc., a usar que se describen en el MCSW.

En un proyecto específico se documentan diversos planes tales como el de las fases del proyecto, el de organización, el de pruebas, el de documentación y el de calidad, etc. En este Plan de Calidad del Producto se definen las características de calidad del producto más relevantes derivadas de la especificación de requerimientos del mismo. Además, en él se determinan los objetivos de calidad para cada característica. Sólo con valores objetivos o definiciones claras para el producto y el proceso de desarrollo es posible planificar, controlar y evaluar la calidad de un producto a través de su ciclo de vida. Incluidas en dicho plan estarán además, los criterios de aceptación, la asignación de recursos y determinación de los jalones de GCSW y el conjunto de actividades de construcción, control, pruebas y evaluación de la calidad.

El Mercado y los contratos determinan:

- Requerimientos Generales de Calidad del Producto
- Requerimientos de GCSW prefijados en el contrato, que pueden estar o no basados

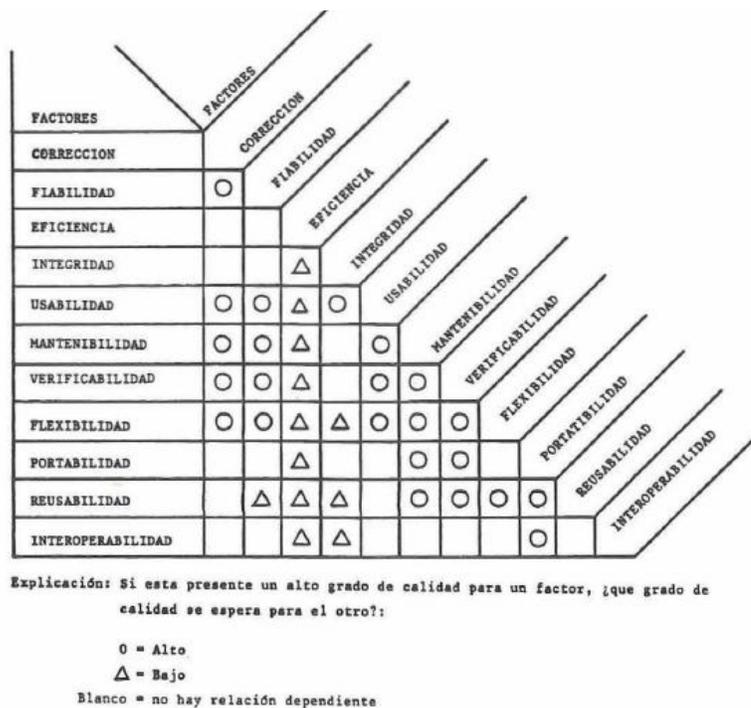


Figura 4
RELACIONES ENTRE LOS FACTORES DE CALIDAD

en alguna norma externa que afectan de Una parte, directamente a la Política de GCSW de la empresa y de otra al Plan de Calidad de cada proyecto.

Así pues la implantación del Plan de GCSW está condicionada por los documentos de la Guía de Aplicación del PGCSW y del Plan de Calidad Especifico de cada proyecto.

En la Guía de Aplicación se describe el proceso que permite definir e instalar el Plan de GCSW en un proyecto específico con la ayuda de la Norma del PGCSW y del Manual de Calidad. Así, define para cada actividad referenciada en la Norma PGCSW qué técnicas, útiles y normas del Manual de Calidad han de ser aplicadas en cada proyecto.

La aplicación del Plan GCSW al proyecto dará como resultado un conjunto de Informes de Calidad procedentes de los procesos de aplicación de dicho plan, los cuales se corresponden con los momentos en que se toman acciones de GCSW a lo largo del ciclo de desarrollo del proyecto.

En la Figura 5 se muestra el esquema de relaciones de las normas, requerimientos y planes de GCSW para la definición e implantación de la GCSW.

NORMAS DE GARANTIA DE CALIDAD SOFTWARE EN USA Y EUROPA

En los últimos años han aparecido distintas normas relacionadas con las actividades y funciones de GCSW. Aparte de sus distintos orígenes, también tienen un enfoque distinto. Así, algunas han sido editadas con el objeto de servir para imponer programas de GCSW en los contratos o para auditoría del cumplimiento de los contratistas respecto a las normas (MIL-STD 1679, MIL-STD-52779A, ANSI/IEEE-STD-730, AQAP-13. FAA-STD-18) mientras otras definen cómo realizarlas (MIL-STD-1670 A). Otras tratan ambos aspectos (RTCA-Do-178, ANS.4.3.2.).

En la Tabla 1 se dan algunas de las normas más difundidas relativas a la GCSW. Es de resaltar por un lado que la mayoría se han creado a partir de 1978 y de otro la preponderancia del ámbito de aplicación militar.

Los militares representan los

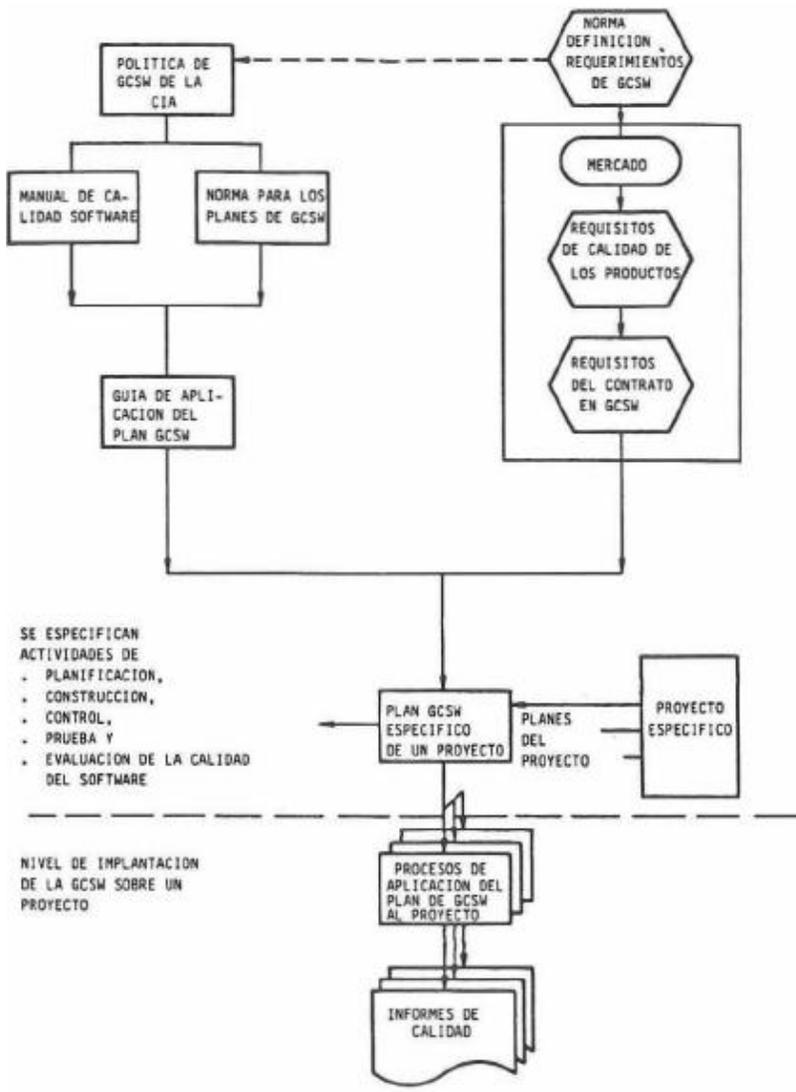


Figura 5 ESQUEMA DE RELACIONES DE LAS NORMAS, REQUERIMIENTOS Y PLANES DE GCSW

requerimientos para organizaciones de usuarios en el área de GCSW para clasificar las obligaciones de los proveedores.

Las civiles tienen un enfoque básicamente para los realizadores interesados en la definición e implantación de la GCSW en sus organizaciones.

Estos dos puntos de vista hacen que haya diferencias entre los dos grupos relativos, en términos generales, a dos aspectos esenciales:

Los objetivos del Plan y el Contexto de implantación, esto es, ámbito, organización, métodos, etc.

Las actividades a realizar, tales como evaluación, recogida y retención de datos de GSW, Formación, Soporte al usuario, etc.

Esta diversidad de normas nos lleva a la necesidad de analizarlas comparativamente para seleccionar cuál de ellas, o un híbrido de las mismas puede servir los intereses de nuestra organización.

TABLA I- ALGUNAS NORMAS RELATIVAS A LA GARANTIA DE CALIDAD DEL SOFTWARE

NORMA	ORIGEN	AÑO	TITULO	AMBITO DE APLICAC.
MIL-STD-52779A	DEP. DE DEFENSA (DOD/USA)	1974	Especificación militar de los Requerimientos del Programa de GCSW	Proveedores del DOD
FAA-STD-018	Administración de la Aviación Federal (USA)	1979	Requerimientos del Programa de Calidad del Software	Aeronáutica
MIL-STD-1679A	DEP. DE DEFENSA (DOD/USA)	1978	Desarrollo software de Sistemas de Armas	Proveedores del DOD
A.N.S. 4.3.2 (Borrador)	IEEE y American Nuclear Society	1979	Criterio normalizado para la aplicación de Sistemas Digitales programables de Centrales de Generación de Energía Nuclear	Centrales de Energía Nuclear
RTCA-DO-178 (EUROCAE-ED-12)	Comisión Radio-técnica para la Aeronáutica	1981	Consideraciones software en sistemas aéreos y y certificación de equipos	Aeronáutica
ANSI/ IEEE-Std-730	IEEE	1981	Norma para los Planes de Garantía de Calidad del Software	Desarrollo y mantenimiento de software crítico
NATO-AQAP-13	OTAN	1981	Requerimientos OTAN para el Sistema de Control de Calidad del Software	Proveedores de la OTAN
GAM-T-17 (Borrador)	Ministerio de Defensa Francés	1982	Guía para la Adquisición Software	Sistemas militares

Esta selección en cualquier caso ha de ser cuidadosa pues de ella dependen recursos y esfuerzos importantes a utilizar una vez decididos a la implantación de la función de GCSW.

PLAN DE GARANTIA DE CALIDAD DEL SOFTWARE

La estructura del Plan aquí presentado está basada en la Norma ANSI/ IEEE Std 730. Esta norma ayuda a la preparación y valoración de los Planes de Garantía de Calidad a implantar en los proyectos software y está dirigida hacia el desarrollo y mantenimiento de software

crítico, esto es, cuando un fallo puede afectar a la seguridad o causar importantes pérdidas sociales o financieras.

Hay tres grupos a quienes interesa la norma: al usuario, al realizador y al público.

El usuario precisa que el producto cumpla los requerimientos identificados en la especificación.

El usuario no tendrá una aptitud de "manos cruzadas" hacia el realizador y relegado solo al final del desarrollo a las pruebas de aceptación.

Si el producto falla, no solo se siguen precisando los mismos requerimientos sino que se habrá perdido una parte del tiempo de desarrollo.

Por tanto el usuario necesita tener un grado de confianza de que el producto está dentro del proceso de adquisición de los atributos requeridos durante el desarrollo software.

El realizador necesita una norma establecida frente a la que planificar y evaluar. No resulta razonable esperar una reorientación completa de proyecto a proyecto.

El público puede verse afectado por la explotación del producto; piénsese en los departamentos de un banco o los pasajeros de ferrocarril que utiliza un sistema de control de tráfico.

Tanto los realizadores como los usuarios que explotan el Sistema pueden verse en situación de demostrar que han actuado de una forma profesional razonable y prudente para garantizar que fueron adquiridos o atributos software que aparecían en los requerimientos.

El objetivo principal de una norma para los Planes de GCSW es de finir las actividades necesarias para producir software de la calidad requerida.

La GCSW se realiza en cada proyecto de acuerdo al escenario pre-establecido con los medios definidos en el Plan de GCSW. En dicho plan se describen las actividades, así como los métodos, técnicas, útiles y normas utilizados y controlar la calidad del SW. Dicho documento tiene:

- La función externa (o contractual) y a que informa al usuario del proceso seguido para garantizar que se cumplen los objetivos de calidad.
- Una función interna ya que de una definición precisa de la GCSW a implantar sobre un proyecto sirviendo así como un documento de referencia y como una guía para el grupo de desarrollo.

La estructura de un Plan de G.C.S. contiene según la norma citada los siguientes apartados que aparecen a continuación:

ESTRUCTURA DEL PLAN DE GARANTIA DE CALIDAD DEL SOFTWARE

- | | |
|--|---|
| 1. INTRODUCCION | 3.3. Revisiones y Auditorias |
| 1.1. Objetivos | 3.4. Gestión de la Configuración |
| 1.2. Documentación de referencia | 3.5. Información de Problemas software y acciones correctivas |
| 2. GESTION | 3.6. Útiles, técnicas y metodologías (solo referencia) |
| 2.1. Organización | 3.7. Control del código |
| 2.2. Tareas | 3.8. Control de los medios |
| 2.3. Responsabilidades | 3.9. Control del proveedor |
| 3. ACTIVIDADES | 3.10. Recogida, mantenimiento y archivo de datos |
| 3.1. Documentación | |
| 3.2. Normas, Técnicas y Convenciones (solo referencia) | |

TENDENCIAS EN GARANTÍA DE CALIDAD DEL SOFTWARE

La implantación de la Garantía de Calidad del Software en la industria general del ramo y en particular se está realizando por etapas, considerando una serie de funciones que se han de adaptar a cada empresa u organismo en particular.

Hay varias facetas que están iniciando su andadura y que van a dar la tónica general en este tema.

1. La calidad se diseña en software, no probada en él.
2. La participación del usuario es cada día mayor.
3. La función de GCSW va a estar implicada en cada proyecto desde sus inicios y participará en cada etapa del desarrollo.
4. GCSW debe ser una función principal e independiente.
5. Cuanto mayor es la dedicación a localizar errores y corregirlos en las primeras fases de desarrollo, tanto más se reducen los costes totales del ciclo de vida de los productos.
6. Hay que abordar el aspecto de formación y selección del personal con las cualificaciones necesarias para formar en cada empresa un grupo eficiente de GCSW.

Los niveles de esfuerzos realizados por las empresas en GCSW pueden agruparse en tres:

- 1) Esfuerzos mínimos, que incluyen Establecimiento de una biblioteca de normas, procedimientos y publicaciones técnicas sobre el tema.
 - a) Creación o adaptación de un conjunto de guías prácticas para su utilización.
 - b) Establecimiento de procedimientos para el control de cambios.
 - c) Revisión de documentos para verificar su nivel de terminación y conformidad de acuerdo a normas, etc.
- 2) Esfuerzos de Nivel Medio, incluye a los del grupo 1 más:
 - a) Establecimiento de procedimientos de gestión de la configuración.
 - b) Revisión de los documentos para verificar su contenido, consistencia y calidad.
 - c) Actuaciones de consulta y supervisión a lo largo del ciclo de desarrollo participando en las documentaciones de desarrollo, control de cambios, repaso, revisiones y auditorías.
- 3) Esfuerzos de Nivel Superior, incluye los realizados en los anteriores más:
 - a) Preparación de un plan de pruebas del Sistema.
 - b) Realización independiente de las pruebas de integración del Sistema.
 - c) Preparación de informes de análisis de pruebas.
 - d) Mantenimiento de una biblioteca de documentos base de configuración, para toda la documentación y código del proyecto.
 - e) Mantenimiento de una biblioteca de pruebas para los planes de pruebas, informes de pruebas y evaluación de técnicas de prueba.
 - f) Seguimiento del estado actual de la tecnología software en relación a las técnicas de prevención de defectos, medidas de calidad del software, útiles de soporte para la GCSW.

CONCLUSIONES

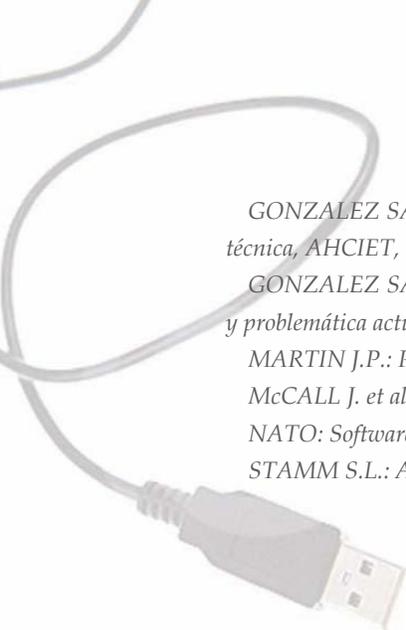
El objetivo de la comunicación ha sido el de mostrar los conceptos de calidad y garantía de calidad del software hoy frente a las necesidades de la industria y usuarios de esos productos. Como conclusiones generales se apuntan las siguientes:

7. La naturaleza de los Sistemas actuales controlados por software determina la necesidad de garantizar unos altos niveles de calidad del software.

8. Se precisan realizar análisis y estudios relativos a las métricas de calidad adecuadas y aceptadas en términos generales por la industria y los usuarios para poder especificar y evaluar de forma objetiva los criterios de calidad fijados en los contratos.
9. El volumen económico previsible del mercado del software y servicios junto con las mayores exigencias de los usuarios van a justificar la orientación de las empresas a acreditar su competencia, sobre todo en la calidad de los productos ofrecidos.
10. La participación en importantes proyectos (en el sentido tecnológico y económico) de dimensión internacional Eureka, ESPRIT, ESA, etc. van a servir de palanca respecto a las prácticas de desarrollo de productos dentro de unos métodos, técnicas y normas de Ingeniería software dirigidas a aumentar la calidad de los productos, mejorar la productividad y controlar sus costes y plazos de entrega.
11. Las empresas de desarrollo han de preparar su propia organización para trabajar con normas de I.S.W. internacionales (ISO, CEI, CCITT, EWICS, IEEE, et c.) o militares (NATO, Ministerio de Defensa, etc.), comenzando por implantar la función de garantía de calidad del software dentro de las mismas e iniciar su participación en las organizaciones profesionales y oficiales para aportar su experiencia, mostrar sus necesidades y ofrecer soluciones para los problemas actualmente planteados en los temas tanto específicos como generales que interesen a la calidad de los productos software.
12. Tanto las industrias como la Administración y los usuarios han de ponerse lo más pronto de acuerdo para poner en marcha el proyecto del Plan Nacional para la mejora de la Calidad avalado por una institución oficial de nivel Nacional de Normalización, Calidad y Homologación actualmente en proyecto y será dentro de dicho Plan donde habrán de ejercerse las actividades correspondientes a la mejora de la calidad de los productos software.

Bibliografía

- ANSI/IEEE: IEEE Standard for Software Quality Assurance Plans. ANSI/IEEE Std 730-1984, Rev. 1
- ARROYO L.: *La vida en un chip*. Espasa-Calpe, Madrid 1985
- BOEHM B.: *Characteristics of Software Quality*. North Holland Pub. Co. 1980
- BUCKLEY F.J.: *A Standard for Software Quality Assurance Plans*. Computer, Vol. 12. Aug. 1979
- DIEBOLD GROUP.: *The Diebold information Technology*, Sean 1983. Diebold Doc. n. 227 T. New York 1984
- DEPARTMENT OF DEFENSE (USA): *Weapon System Software Development*. MIL-S-1679, Dec. 1978
- IDEM: *Military Specification Software Quality Assurance Program Requirements*. MIL-S- 52779 A, 1 Aug. 1979 (MIL-S-52779, 5 Apr. 1979, 1ª versión)
- DUNN R.H. ULLMAN R.S.: *Quality Assurance for Computer Software*. Electrical Communication, Vol 57, n, 4, 1983
- FEDERAL AVIATION AGENCY: *Computer software Quality Program Requirements*. Dept. of Transportation (USA) FAA-STD-018, May 1977
- GONZALEZ SANZ J.: *La calidad de los productos software*. Comité de Calidad del Software, de la Asociación Española para el Control de la Calidad, DOC-CS-AECC/1, Madrid 1983
- GONZALEZ SANZ J.: *Presentación del Comité de Calidad del Software de la A.E.C.C.* PRE-CS-AECC, n. 6, Ver. 01, 15/1/84 (CTNE)
- GONZALEZ SANZ J.: *Informe de la Segunda Reunión del CCSW de la EOQC*. Toulouse, Octubre 1983; INF-CS-AECC/2, Madrid 1983
- GONZALEZ SANZ J.: *Informe de la Tercera Reunión del CCSW de la EOQC*. Brighton, Junio 1984; INF-CS-AECC/3, Madrid 1984



GONZALEZ SANZ J.: *Estudio de las tendencias del Software de los Sistemas de Comunicación de datos* Contribución técnica, AHCJET, COM-C 22/83 Santiago de Chile, Noviembre 1983

GONZALEZ SANZ J.: *El software de los sistemas de comunicaciones de datos: características, estructura, actividades y problemática actual.* Contribución técnica, AHCJET, COM-C 47/85 Lima, Perú, Abril 1985

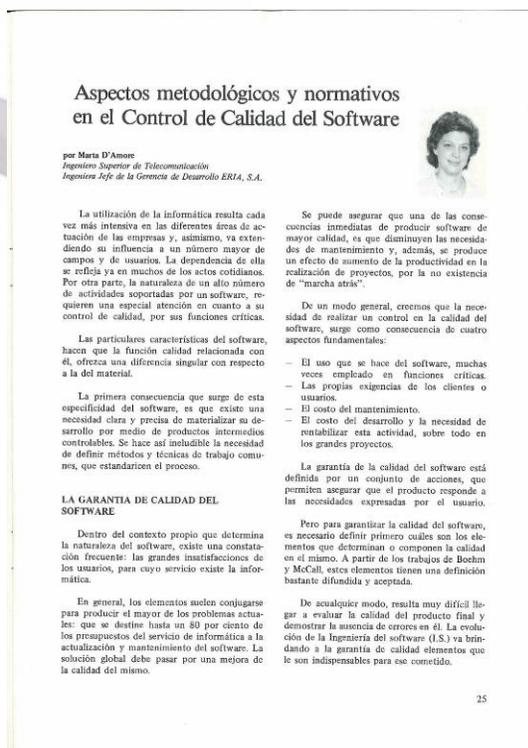
MARTIN J.P.: *Pour un standard du plan qualite du logiciel.* *Rv. AFCIQ*, Vol XIX, n. 2, Juin 1983

McCALL J. et al.: *Factors in Software Quality* 3 Vols, NT/S AD-AD49- 0/4, 015, 155, Nov. 1977

NATO: *Software Quality Control System Requirements.* AQAP-1 3, NATO, Aug. 1981

STAMM S.L.: *Assuring Quality -Quality Assurance, Datamation*, March 1981

Aspectos metodológicos y normativos en el Control de Calidad del Software



Autor: Marta D'Amore
Ingeniero Superior &
Telecomunicación Ingeniera
Jefe & 14 Gerencia de
Desarrollo ERIA, S.A.



Publicado en el

La utilización de la informática resulta cada vez más intensiva en las diferentes áreas de actuación de las empresas y, asimismo, va extendiendo su influencia a un número mayor de campos y de usuarios.

La dependencia de ella se refleja ya en muchos de los actos cotidianos. Por otra parte, la naturaleza de un alto número de actividades soportadas por un software, requieren una especial atención en cuanto a su control de calidad, por sus funciones críticas.

Las particulares características del software, hacen que la función calidad relacionada con él, ofrezca una diferencia singular con respecto a la del material. La primera consecuencia que surge de esta especificidad del software, es que existe una necesidad clara y precisa de materializar su desarrollo por medio de productos intermedios controlables. Se hace así ineludible la necesidad de definir métodos y técnicas de trabajo comunes, que estandaricen el proceso.

La Garantía de Calidad del Software

Dentro del contexto propio que determina la naturaleza del software, existe una constatación frecuente: las grandes insatisfacciones de los usuarios, para cuyo servicio existe la informática.

En general, los elementos suelen conjugarse para producir el mayor de los problemas actuales: que se destine hasta un 80 por ciento de los presupuestos del servicio de informática a la actualización y mantenimiento del software. La solución global debe pasar por una mejora de la calidad del mismo.

Se puede asegurar que una de las consecuencias inmediatas de producir software de mayor calidad, es que disminuyen las necesidades de mantenimiento y, además, se produce un efecto de aumento de la productividad en la realización de proyectos, por la no existencia de "marcha atrás". De un modo general, creemos que la necesidad de realizar un control en la calidad del software, surge como consecuencia de cuatro aspectos fundamentales:

- El uso que se hace del software, muchas veces empleado en funciones críticas.

- Las propias exigencias de los clientes o usuarios
- El costo del mantenimiento.
- El costo del desarrollo y la necesidad de rentabilizar esta actividad, sobre todo en los grandes proyectos.

La garantía de la calidad del software está definida por un conjunto de acciones, que permiten asegurar que el producto responde a las necesidades expresadas por el usuario.

Pero para garantizar la calidad del software, es necesario definir primero cuáles son los elementos que determinan o componen la calidad en el mismo. A partir de los trabajos de Boehm y McCall, estos elementos tienen una definición bastante difundida y aceptada.

De cualquier modo, resulta muy difícil llegar a evaluar la calidad del producto final al y demostrar la ausencia de errores en él. La evolución de la Ingeniería del software (I.S.) va brindando a la garantía de calidad elementos que le son indispensables para ese cometido.

Ello se debe a que la I.S. es un conjunto de métodos, técnicas y herramientas, utilizadas para desarrollar y mantener el software, cuyo objetivo fundamental es el de mejorar tanto la calidad, como la productividad.

Así como otros sectores de la economía se automatizan para aumentar la productividad y mejorar los trabajos, la informática también necesita de herramientas software que conllevan a una mejora de todo el entorno de realización de proyectos, sobre todo en el marco actual de costes. Pero, hasta el presente, en lo que respecta a las herramientas en el mercado, estas son aún insuficientes para algunos aspectos y, en otros, presentan un panorama tan heterogéneo que impiden obtener la integración indispensable entre los resultados aportados por cada una. Con el agravante de que los costes de formación y utilización, aún suelen ser elevados.

Debemos sin embargo notar que, el aporte del marco metodológico y de técnicas que nos está ofreciendo la I.S., ha permitido ir racionalizando y formalizando los trabajos, abandonando aquella etapa puramente artesanal de los comienzos, en la que cualquier verificación se realizaba una vez obtenido el producto final.

Evidentemente, la garantía de calidad de un producto software terminado, se obtiene garantizando la calidad de su desarrollo. Para ello, como decíamos, cada actividad se debe materializar por algún producto intermedio o documento, sobre el que sea posible efectuar controles.

Así, la garantía de calidad del software, no se apoya sólo en poder medir la bondad del funcionamiento de los programas. Necesita de un entorno normalizado de trabajo que cubra un espectro mucho más amplio de elementos; la gestión de proyectos; la metodología de trabajo; la gestión de la configuración y la propia gestión de la calidad.

Gestión de Proyectos

El primer principio básico que debe cumplirse para poder garantizar la calidad del software, es que se realice una descomposición del proceso total de su desarrollo, tanto desde el punto de vista técnico, como metodológico. Es decir: que el producto final quede descompuesto en elementos medibles y que su desarrollo esté estructurado en actividades controlables, soportadas por la definición metodológica de un ciclo de vida.

Será además necesario que queden establecidas claramente las responsabilidades de todos los participantes involucrados en el proyecto de desarrollo de software. No sólo en cuanto a su carga de trabajo, sino también en lo relativo a su función específica (quién ejecuta actividades de producción; quién dirige; quién controla; quién asesora; etc.).

Ya se ha mencionado asimismo que, un componente de la calidad importante, es que no se

sobrepasen costos y se cumplan los plazos. Por tanto, se hace imprescindible que se midan periódicamente los avances del proyecto, y se los compare con los valores dados en la planificación de la puesta en marcha, y los sucesivos.

Todas estas consideraciones forman parte de la función "planificación y control de proyectos". Función que tendrá que estar soportada por un método en el que se defina la estructura de un sistema de información, por medio del cual se pueda conocer el estado del proyecto en todo momento. Método que también deberá permitir visualizar las diferentes responsabilidades que se dan dentro del flujo de información.

Metodología en la Producción del Software

La Ingeniería del Software nos brinda una aproximación sistemática al desarrollo, explotación y mantenimiento del software. El axioma de partida en esta sistematización es la definición de un ciclo de vida para el software, esencial para la descomposición metodológica del proceso de producción que exigen las acciones de calidad, como antes mencionáramos.

Presentamos como ejemplo de referencia en este artículo, el ciclo de vida normalizado por IEEE. Quedando claro que en el entorno diario de trabajo se pueden encontrar distintos nombres, así como pequeñas modificaciones en las fronteras, Lo esencial es cumplir con el esquema cronológico, y que se produzcan los resultados y documentos correspondientes a cada grupo de actividades homogéneas (especificaciones; diseños; codificación;...). El ciclo mencionado define las siguientes fases:

- Especificaciones
- Diseño
- Programación
- Pruebas
- Instalación y entrega
- Explotación y mantenimiento
- Cancelación (eventualmente)

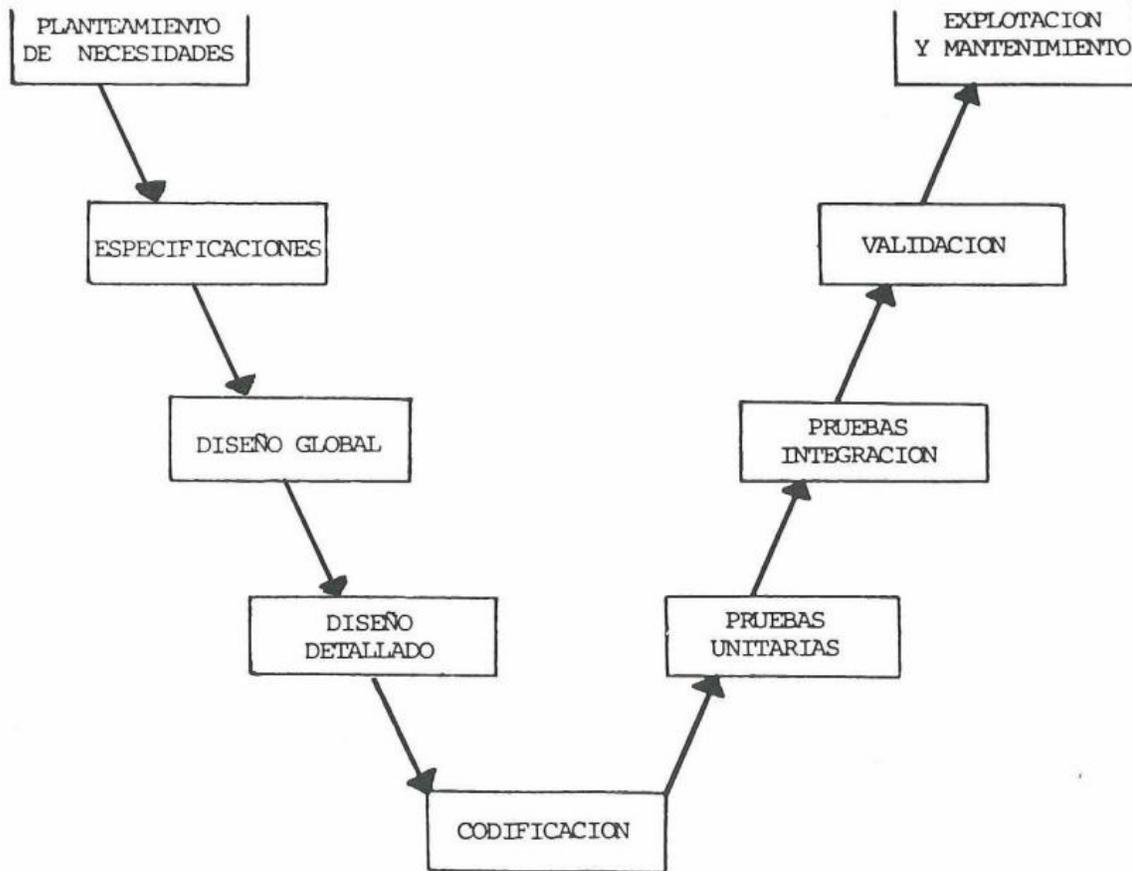
Estas fases suelen estar normalmente precedidas por unos estudios previos o un plan informático.

Una forma más idónea de representar el ciclo de vida, para poner claramente de manifiesto las acciones de calidad, es la clásica forma en "V" (Figura 1).

En cada una de las fases, o para algunos conjuntos de actividades, es factible utilizar técnicas y herramientas. Así, podría utilizarse una técnica de programación estructurada para el diseño y codificación de programas, o una herramienta como los diccionarios de datos, para ayuda en el diseño.

La integración de los procedimientos técnicos y de gestión que estén relacionados entre si y sean factibles de utilizar en el desarrollo y mantenimiento del software, definen a una metodología. Una metodología de calidad, por tanto, debe ser capaz de normalizar el proceso de producción, y de integrar en él los elementos que hagan posible mejorar y disciplinar la realización de las diferentes actividades.

Si la garantía de calidad del software depende en buena medida de las verificaciones sistemáticas que se hagan de los trabajos que se vayan realizando, la metodología le aporta sus elementos indispensables. Por ejemplo, asegurando la legibilidad de los programas y documentos, o brindando los elementos para que puedan reducirse las actividades futuras de mantenimiento.



(Figura 1)

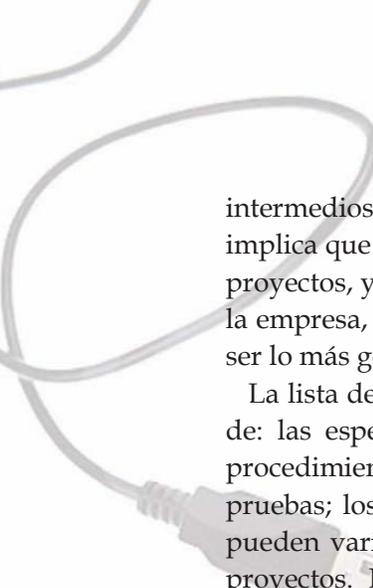
Para ello, la estructura de una metodología, creemos que deberá aportar siempre:

- Las normas para realizar cada una de las actividades de las fases.
- La definición del resultado de esas actividades, y de la estructura y contenido de la documentación a elaborar, así como las reglas para incorporar las modificaciones en versiones sucesivas.
- La definición de los documentos estandarizados a utilizar como soporte para el trabajo.
- El establecimiento de las técnicas, herramientas y ayudas que puedan emplearse en cada tarea o conjunto de tareas.
- La flexibilidad y modularidad suficiente, como para permitir la evolución y uso de nuevas técnicas y herramientas, que mejoren la calidad y productividad.

Gestión de la Configuración

Gestionar la coherencia de todos los elementos que se van definiendo y creando a lo largo de la vida de un pequeño proyecto, es fácil. Pero esa gestión se complica cuando se está en presencia de proyectos más voluminosos, críticos o complejos. En ese caso y ante la ausencia de la función gestión de la configuración, puede darse que no se sepa en un momento dado dónde se encuentra un programa fuente, o aún más corriente que no se tengan a un mismo nivel la documentación funcional y la versión en que se encuentra el software desarrollado.

Por eso se hace necesario que se gestionen de manera racional todos los productos



intermedios que se vayan generando a lo largo del ciclo de vida de un proyecto software. Ello implica que la actividad "gestión de la configuración" tenga que aparecer como otra más en los proyectos, ya sea como función ejercida por el grupo responsable de la calidad del software en la empresa, ya sea como una función realizada por el propio equipo de desarrollo, como suele ser lo más generalizado.

La lista de elementos de configuración puede ser muy amplia y se refiere a los componentes de: las especificaciones; las documentaciones del diseño; los códigos fuentes y objeto; los procedimientos de explotación; los manuales de usuario y explotación; las documentaciones de pruebas; los informes de errores; los resultados de acciones de calidad; etc. Estos elementos pueden variar muchísimo en cuanto a su complejidad, tipo o volumen, según los diferentes proyectos. En general, es una función de la propia gestión de la configuración llegar a identificar y a definir los elementos de configuración de un proyecto.

El conjunto de elementos coherentes y completos que se aprueban al finalizar una fase del ciclo de vida, conforma un referencial ("base line"). Sobre los referenciales se realizan las acciones formales de control de calidad, quedando "congelados" a partir de esos controles. Por lo tanto, representan la base para cualquier tratamiento futuro de modificaciones, y cualquiera de ellas debería implicar una petición formal de actualización del referencial.

Se puede así definir una configuración de un proyecto software, como el conjunto completo y coherente de elementos que exista en cada instante del ciclo de vida, relacionado con una configuración especial que es el referencial. Por ejemplo: la configuración versión 4 de los programas, que corresponde a la versión 2 de las especificaciones, y a la versión 2 del diseño de un proyecto.

La gestión de la configuración de un proyecto, es una actividad que deberá materializarse a través de cuatro funciones básicas (definición IEEE):

- Identificar y definir los elementos de configuración de un sistema.
- Controlar las modificaciones que se produzcan durante todo el ciclo de vida y difundirlas.
- Registrar las modificaciones e informar sobre el estado de la configuración de manera periódica y puntual.
- Verificar la completitud y corrección de los elementos y de sus configuraciones.

La consecuencia de ejercer y realizar estas funciones, redundan en un beneficio importante dentro de la producción del software.

En primer lugar, permite hacer "visible" su proceso de desarrollo. En segundo lugar, puede facilitar la reutilización de los productos, hecho cada vez más importante en los desarrollos. Por último, al controlar las modificaciones y asegurar la coherencia de los componentes del software, participa de manera activa en la garantía de calidad del software, suministrándole los soportes idóneos para realizar las acciones de calidad.

Es evidente que la gestión de la configuración, se nutre y adquiere existencia a partir de la utilización de una metodología de desarrollo, que es quien le irá brindando los elementos normalizados a gestionar.

Gestión de la Calidad

La necesidad de garantizar la calidad del software, implica, en general, la aparición dentro de la organización de la empresa, de un equipo con determinadas funciones. La responsabilidad de las actividades involucradas en la garantía de calidad, recaen tanto en este equipo de calidad, como en los propios realizadores.

La gestión de la calidad de un proyecto software se apoya en dos elementos: un conjunto de documentos y un conjunto de actividades.

Con respecto a los documentos, dos son los que soportan fundamentalmente la gestión de la calidad: el Manual de Calidad y el Plan de Garantía de Calidad del Software (PGCS). En otro artículo de este monográfico, se describen con detalle estos documentos y los esfuerzos de normalización con respecto a ellos.

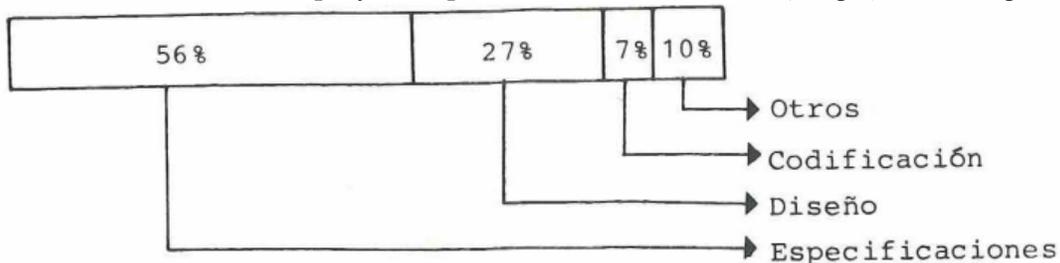
Decíamos que el segundo elemento sobre el que se apoya la gestión de calidad, es el de las propias actividades orientadas a su control y garantía.

El núcleo de estas actividades está orientado a la verificación sistemática de la calidad de los trabajos que se están desarrollando dentro del ciclo de vida. Estas verificaciones deben hacerse en varios sentidos:

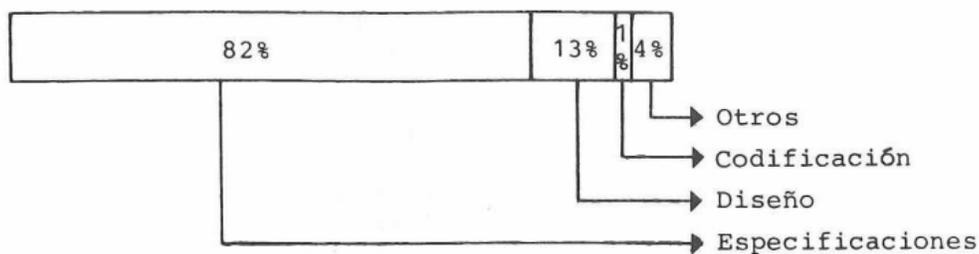
- Verificación del cumplimiento formal de las reglas, normas o estándares fijados.
- Verificación de la coherencia entre los resultados de las diferentes etapas.
- Verificación de la conformidad entre el producto y su definición.

Es evidente que bajo esta óptica, se abandonan las viejas prácticas de desarrollar y luego probar los programas al final del desarrollo. En función de la calidad es necesario que cada fase del ciclo de vida sea verificada y aceptada, antes de comenzar la siguiente, para no contribuir a una producción en cadena de los errores y dificultar su identificación.

Según las estadísticas, en un proyecto tipo la distribución de fallos ("bugs") sería la siguiente:



Mientras que los esfuerzos para encontrar y corregir esos fallos, representaría los siguientes valores:



Es indudable que por razones de productividad, también se hace necesario encontrar cuanto antes los errores iniciales, que son los más caros para corregir.

Las verificaciones durante el ciclo de vida, deberían entonces responder al esquema que se ha marcado en la Figura 2.

En esta representación del ciclo de vida, se pueden detectar tres tipos de controles de calidad:

- Relectura de documentos
- Verificaciones con respecto a las etapas precedentes
- Pruebas

Estos suelen ser controles típicos ejecutados por el equipo de desarrollo. Serán función del

equipo de calidad verificar que las actividades de control previstas se han llevado a cabo, y que sus resultados y el estado de la documentación permiten pasar a la actividad siguiente.

Este conjunto de verificaciones facilitaría el descubrimiento de errores; reduciría las vueltas atrás paralizantes y mejoraría en consecuencia la calidad.

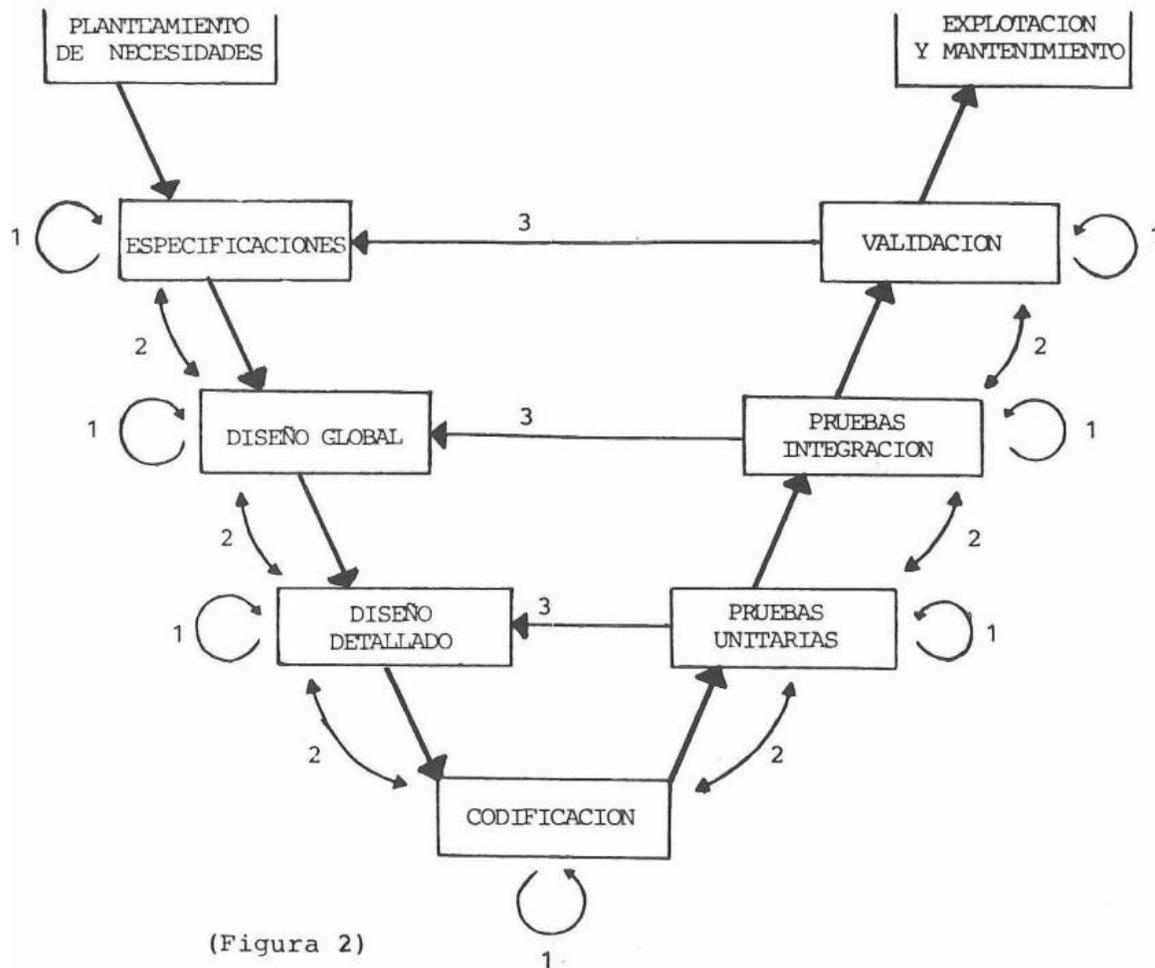
Los métodos a emplear en las acciones de calidad, son esencialmente cuatro: revisiones lecturas cruzadas; auditorías, inspecciones; pruebas.

Las pruebas estarán normalmente soportadas por una planificación y una documentación, estructuradas por la metodología de trabajo. El rol específico de la garantía de calidad, será el de asegurar que la actividad de pruebas está planificada; su documentación es la idónea y que las pruebas previstas se han realizado con los resultados correctos.

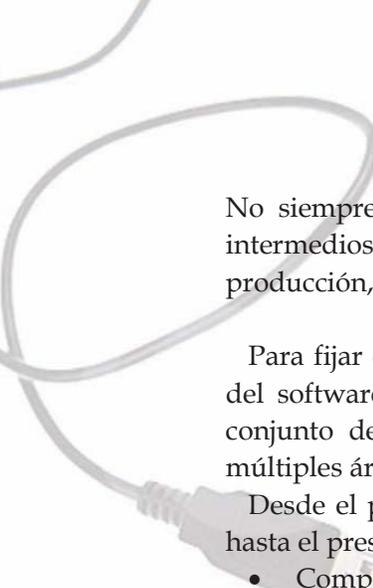
El PGCS de cada proyecto fijará, por otro lado, cuáles van a ser los criterios de aceptación y de finalización de las diferentes pruebas. Es de notar, por último, que las acciones de calidad pueden hacer uso de diferentes herramientas software: editores de texto; analizadores lexicográficos; etc., así como de otras ayudas basadas en la evaluación o medida cuantitativa de la calidad.

EVALUACION DE LA CALIDAD DE SOFTWARE

Uno de los elementos fundamentales a utilizar en el entorno de las acciones de calidad del software, es el de la utilización de técnicas o herramientas de medición o evaluación.



(Figura 2)



No siempre es posible llegar a cuantificar exactamente la calidad de todos los productos intermedios. Depende además, en muchos casos, de que se utilicen estándares y normas en la producción, de manera que se puedan recoger los parámetros necesarios para la evaluación.

Para fijar criterios, definimos a una métrica como la medida cuantitativa de una propiedad del software, en la que se hacen intervenir una o más medidas directas. La metrología -o conjunto de técnicas que nos permiten realizar esas medidas- se aplica evidentemente a múltiples áreas (gestión de proyectos, auditoría de la explotación, etc.).

Desde el punto de vista específico de la calidad, el soporte de la metrología está orientado hasta el presente fundamentalmente hacia tres características:

- Complejidad
- Fiabilidad
- Factores de calidad

La metrología de la complejidad se orienta fundamentalmente a la mantenibilidad del software. Está basada en el análisis estático del código de los programas, tomando como axioma de partida que la complejidad es una característica de influencia negativa para el software.

Entre las medidas de complejidad propuestas, las más conocidas son las que miden la complejidad del texto de los programas, o las que se dedican a su complejidad estructural. Relacionada con la primera, destaca la técnica de Halstead, que sirve más para constatar la complejidad del software, que para prevenirla.

T. McCabe, ha realizado una modificación de esta técnica, basada en la teoría de grafos y en la medida de la complejidad estructural a través de los IF y GO TO de los módulos o programas.

Estas técnicas necesitan en general herramientas software que las soporten, lo que provoca ciertos trastornos debido a la diversidad de lenguajes existentes.

La metrología de la fiabilidad, se basa en la observación de los programas ya en funcionamiento, por tanto se aplica a las fases de pruebas y, fundamentalmente, a la de explotación. Esta aproximación está todavía en un estado de evolución, y necesitará de mayores desarrollos. De cualquier manera se pueden citar ya vanos modelos conocidos: Musa, Goel-Okumoto, Jelinski-Moranda.

Las medidas en este caso son de tipo dinámico y sobre el producto, es decir, sobre los fallos observados. Casi todos los trabajos citados, se basan en una modelización de los fallos, por medio de una distribución estadística. Dado que será prácticamente imposible poder obtener un software que no tenga absolutamente ningún error, para los proyectos críticos (aeronáutica, espacial) se hace imprescindible llegar a cuantificar un porcentaje de fallos probables y aceptables. Este valor, servirá asimismo como criterio para dar por finalizadas las pruebas del producto y su certificación.

La metrología de los factores de calidad, tiene por origen los trabajos de Boehm y McCall. Se basa en un examen del producto ya en proceso, pero también en un análisis estático de las especificaciones, el diseño y la programación. Por tanto, se puede usar más para gestionar y prever la calidad que para constatarla, y es más global que las dos anteriores.

Las medidas o métricas, en este caso, se aplican sobre los programas y la documentación. En general son métricas binarias (respuestas "si" o "no") o ratios. Por ejemplo, una métrica del

criterio "coherencia", perteneciente al factor "corrección", podría ser la respuesta a: "¿se usa un método normalizado de Diseño?" o "En qué % se respetan las convenciones de símbolos".

La metrología de los factores de calidad pueden, indudablemente implantarse en fases más tempranas que las otras dos mencionadas. Es también mucho más manual -aunque podrían emplearse algunas herramientas- y puede incluir a la de complejidad.

Cualquiera de estas metrologías puede combinarse con las otras. Lo esencial será adaptarlas a las reales necesidades de calidad del proyecto.

Evolución de la Ingeniería del Software

Como ya se ha mencionado, la evolución de la I.S. irá brindando un soporte cada vez más importante a la garantía de su calidad. Hasta el presente existe, un conjunto amplio de técnicas y herramientas que asisten a las diferentes actividades involucradas en un proyecto. Nos encontramos así con herramientas software directamente relacionadas con las diferentes fases del ciclo de vida.

Por otro lado, existe un grupo de herramientas cada vez más numeroso relacionadas con actividades tales como: la documentación; la gestión de proyectos; la gestión de la configuración; la gestión de la calidad; la metrología. El mercado ofrece ejemplos de herramientas en cualquiera de estas áreas.

La elección de las herramientas óptimas para mejorar la calidad y la productividad, suele ser una tarea delicada, debido -entre otros factores- a la falta de integración entre las diferentes herramientas y sus resultados. La primera regla a contemplar, sería la de esforzarse por utilizar herramientas compatibles.

La evolución está llevando al concepto de entornos integrados en los que se tendrá un grupo de herramientas que cubran la totalidad de actividades, y que puedan ser utilizadas conjuntamente en la realización de un proyecto software.

Todavía pasará algún tiempo hasta esa integración total, pero es algo a lo que se tiende en pos de la reutilización de los componentes software y pensando en una industrialización de la producción a través de la automatización del proceso. Aunque habrá que ver si se consigue una estandarización que haga compatibles los entornos

Conclusiones

La garantía de calidad del software permite reducir los riesgos existentes en su desarrollo y posterior utilización. No debe pensarse que los controles de calidad sólo pertenecen al terreno de los grandes proyectos o a los proyectos críticos.

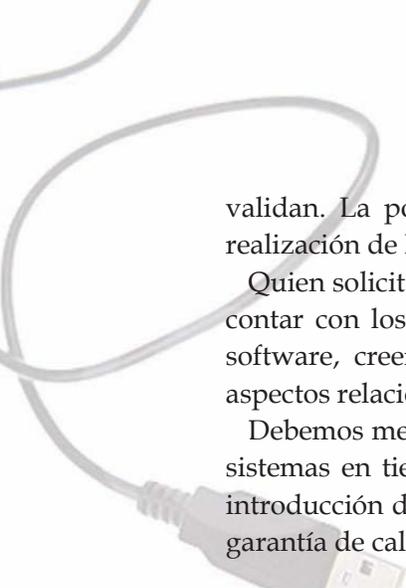
En cualquier entorno de desarrollo, deberán establecerse siempre estándares y métodos de trabajo, adaptados al contexto real, que permitan minimizar los diferentes riesgos inherentes a la naturaleza del software (insatisfacciones, costos, seguridad,...).

Cuando las características de los proyectos software lo requieran, deberá materializarse, dentro de la organización de las empresas, la presencia de un equipo de calidad del software.

La primera regla de oro, debería ser la delimitación estricta de responsabilidades entre este equipo y el de desarrollo. La segunda, la responsabilidad en la formación en calidad que adquiere ese equipo concreto.

Por otro lado, la garantía de la calidad del software tendrá un costo real y directo, indudablemente rentabilizado por los beneficios obtenidos del ejercicio de esa función.

Como elemento indispensable en este programa, el usuario o cliente tendrá que saber expresar sus necesidades, pues para ellas se construyen las aplicaciones y sobre ellas se



validan. La posible existencia de cláusulas contractuales de calidad, serán la base para la realización de la planificación de la calidad de un producto, expresada a través de un PGCS.

Quien solicita un trabajo, debe conocer y fijar sus cláusulas de calidad. Quien lo realiza, debe contar con los medios para garantizarla. Desde la perspectiva de una empresa de servicios software, creemos fundamental poseer el bagaje que nos permita satisfacer los diferentes aspectos relacionados con la calidad.

Debemos mencionar por último que, la evolución de la aplicación de la informática (grandes sistemas en tiempo real, mayor complejidad y condiciones críticas, distribución de recursos, introducción de la inteligencia artificial,...) irá perfilando la propia evolución y actuación de la garantía de calidad del software.

Referencias

J. GOURGEOIS. *Ateliers de Génie Logiciel: état de l'art et perspectives* (Reme de Genie Logiciel 1985)

R.L. GLASS. *Software Reliability Guidebook* (Prentice-Hall 1979)

IEEE. *Standard for Software Quality assurance plans* (730 - 1984)

G. MYERS. *El arte de probar el software*. (El Ateneo. 1983)

La Formación: Una clave esencial en la Mejora de la Calidad del Software

La Formación: Una clave esencial en la Mejora de la Calidad del Software

Por Julio González Sanz
Presidente del Comité de Software de la A.E.C.C.
(presentado en el Congreso IBERICO de la Calidad, Estoril, Noviembre 1988).

RESUMEN.— En esta contribución se muestra la necesidad de formación que tienen las áreas de gestión, calidad y desarrollo software para llevar a cabo el objetivo de calidad de los productos software que ofrece la empresa al mercado.

En la actualidad no se imparte a nivel de Universidad ninguna especialización en ingeniería y calidad de software, por lo que la industria precisa introducir esta formación de forma coherente y continuada, si quiere alcanzar sus objetivos de calidad, productividad y costes.

Se presenta un modelo conceptual jerarquizado para la formación profesional de aplicable a organizaciones que quieren organizar un servicio de formación en ingeniería y calidad del software. Se incluyen las disciplinas que deben cubrirse para abordar una visión general e integrada de todos los aspectos que intervienen a la hora de generar software de calidad.

INTRODUCCION.— Hoy más que nunca, la calidad se ha convertido en la clave del marketing internacional y nacional de los productos industriales. Para alcanzar los requisitos cada vez más altos de calidad exigidos por los usuarios de la industria software, deben establecerse y mantenerse sistemas estrictos de aseguramiento y de control de calidad, además de construir la calidad del producto a través del diseño. Esto es especialmente cierto en el caso del software cuya realización es conceptualmente y en la práctica un producto resultado total del diseño y no de la fabricación como ocurre con el resto de los productos de naturaleza física.

16

Por ello las industrias software deben prepararse en los próximos tiempos para poder hacer frente a dichas exigencias de calidad a través de procesos de mejora de la misma.

Los mayores exponentes dentro del mundo profesional de la calidad como Juran, Deming y Crosby ponen el énfasis en la educación y formación como componentes estratégicos clave para la mejora de la calidad, aunque con diferentes enfoques. Juran apunta a las prácticas de gestión de la calidad y a las técnicas de resolución de problemas.

De otra parte proporciona un enfoque de sistemas para el Control de la Calidad y la mejora de todas las partes de la organización.

Deming pone el énfasis en las técnicas estadísticas. La formación de Crosby está dirigida hacia el desarrollo de una nueva cultura de la Calidad e implantación del proceso de mejora de la misma.

La mejora de la calidad del software debe concentrarse en la mejora de la calidad del proceso que genera el producto. De otro lado, la compañía debe asegurar la mejora de la calidad a lo largo de la organización. Esto incluye no solo el desarrollo, sino también las áreas de gestión administrativa, de personal, de marketing., etc.

Conceptualmente la formación es un servicio que debe proporcionarse para satisfacer las necesidades de los usuarios frente a unas necesidades bien definidas.

Boletín CALIDAD núm. 1.— Enero-Febrero 1989

Autor: Julio González Sanz Presidente del Comité de Software de la A.E.C.C. (presentado en el I Congreso IBERICO de la Calidad Estoril, Noviembre 1988).

Publicado en el Boletín Calidad núm. 1.- Enero-Febrero 1989

RESUMEN.

En esta contribución se muestra la necesidad de formación que tienen las áreas de gestión, calidad y desarrollo software para llevar a cabo el objetivo de calidad de los productos software que ofrece la empresa al mercado. En la actualidad no se imparte a nivel de Universidad ninguna especialización en ingeniería y calidad de software, por lo que la industria precisa introducir esta formación de forma coherente y continuada, si quiere alcanzar sus objetivos de calidad, productividad y costes. Se presenta un modelo conceptual jerarquizado para la

formación profesional de aplicable a organizaciones que quieran organizar un servicio de formación en ingeniería y calidad del software. Se incluyen las disciplinas que deben cubrirse para abordar una visión general e integrada de todos los aspectos que intervienen a la hora de generar software de calidad.

INTRODUCCIÓN.

Hoy más que nunca, la calidad se ha convertido en la clave del marketing internacional y nacional de los productos industriales. Para alcanzar los requisitos cada vez más altos de calidad exigidos por los usuarios de la industria software, deben establecerse y mantenerse sistemas estrictos de aseguramiento y de control de calidad, además de construir la calidad del producto a través del diseño. Esto es especialmente cierto en el caso del software cuya realización es conceptualmente y en la práctica un producto resultado total del diseño y no de la fabricación como ocurre con el resto de los productos de naturaleza física.

Por ello las industrias software deben prepararse en los próximos tiempos para poder hacer frente a dichas exigencias de calidad a través de procesos de mejora de la misma.

Los mayores exponentes dentro del mundo profesional de la calidad como Juran, Deming y Crosby ponen el énfasis en la educación y formación como componentes estratégicos clave para la mejora de la calidad, aunque con diferentes enfoques. Juran apunta a las prácticas de gestión de la calidad y a las técnicas de resolución de problemas.

De otra parte proporciona un enfoque de sistemas para el Control de la Calidad y la mejora de todas las partes de la organización.

Deming pone el énfasis en las técnicas estadísticas. La formación de Crosby está dirigida hacia el desarrollo de una nueva cultura de la Calidad e implantación del proceso de mejora de la misma.

La mejora de la calidad del software debe concentrarse en la mejora de la calidad del proceso que genera el producto. De otro lado, la compañía debe asegurar la mejora de la calidad a lo largo de la organización. Esto incluye no solo el desarrollo, sino también las áreas de gestión administrativa, de personal, de marketing., etc.

Conceptualmente la formación es un servicio que debe proporcionarse para satisfacer las necesidades de los usuarios frente a unas necesidades bien definidas.

LA CALIDAD DEL SOFTWARE Y SU ENTORNO.

Desde hace solo unos pocos años se viene reconociendo que el diseño, el desarrollo y el mantenimiento de sistemas software es cada vez más caro, tiene más riesgos y es más impredecible que en el caso del equipo físico (hardware).

Las técnicas para gestionar el proceso de desarrollo para conseguir calidad hasta ahora razonablemente establecidos: sin embargo su introducción en las organizaciones presentan serias dificultades.

- Al grupo de calidad deben pertenecer solo aquéllos que tengan la formación y experiencia suficiente para comprender lo que es el proceso de diseño y desarrollo de los sistemas software. Deben ser capaces de realizar análisis de diseño y pruebas. No caben los cantamañanas y los picaliendres.
- La calidad del diseño no se puede mandar o realizar sin auditorías o inspecciones. Hay que evitar el papel exclusivo de seguimiento para verificar el cumplimiento a normas. Por ello el grupo debe situarse dentro de la organización de diseño.
- Hay que establecer rápidamente credibilidad, apuntando el trabajo inicial sobre un área donde se consiga un alto provecho visible con la menor perturbación sobre la marcha del proyecto. Hay que evitar la tendencia a actuar como un Grupo de depuración. Los diseñadores son responsables de la validación y la ejecución de las pruebas. - Es necesario revisar las prácticas internas de desarrollo y educar a la dirección sobre los agentes principales de los costes del software.
- Hay que tener suficiente paciencia para ir realizando los cambios necesarios, por lo que hay que establecer ciertas etapas de introducción de la calidad en la empresa. Los hábitos de trabajo no se cambian por decreto.
- Transferir las experiencias a otros grupos de diseño conectando en nuevos aspectos aún no tratados. Esto requiere nuevas experiencias y conocimientos a medida que evoluciona el grupo.
- No dar soluciones genéricas. Se deberán dar soluciones hechas a medida del negocio, de su dimensión y de su cultura corporativa.
- Demostrar con todo lo anterior a los diseñadores y a los directivos escépticos que la calidad del software es rentable y reporta beneficios tangibles.

La calidad del producto final está relacionada directamente con:

1. La tecnología utilizada.
2. La formación y capacidad técnica del personal, Y
3. Los controles de gestión, aplicados a todo el proceso.

Estos últimos dependen de las normas aplicadas al proyecto junto con el programa de calidad realizado bajo estas normas, hacen que se consiga la deseada calidad del producto final.

De otra parte existe una conexión directa entre las mejoras de calidad y mejoras de productividad.

La calidad del software solo se puede producir con personal experimentado, motivado y tecnológicamente actualizados. Como complemento indispensable, herramientas de diseño y prueba, que no bastan si no se implanta una metodología de desarrollo que gobierne el proceso de calidad. Además la Calidad del Software sólo puede llevarse a cabo por medio de una gestión del desarrollo adecuada.

Resulta esencial la adherencia a una metodología de diseño disciplinada que garantice la calidad deseada. Esto suele significar un cambio en la cultura de diseño y en los hábitos de trabajo.

En algunos casos las herramientas pueden ser comunes, aunque su uso dependerá del punto de vista de la función que las utilice.

A la función de control de la calidad le corresponde la realización de medir y verificar los procesos y productos. A la función de aseguramiento de la calidad le corresponde la realización de normas y procedimientos, lo que representa la base de conocimientos de la empresa.

El enfoque dado en esta ponencia es el de la formación profesional impartida fuera de los ciclos universitarios ordinarios, esto es para posgraduados con cierta experiencia empresarial en diseño y desarrollo software o en el área de calidad.

La formación profesional en calidad del software debe tener entre sus objetivos los dos siguientes, que consideramos fundamentales:

1. La preparación de personal de alta cualificación.
- 2.- Actualización continua del personal en nuevas técnicas y procedimientos.

LA CALIDAD DEL SOFTWARE Y LA FORMACIÓN.

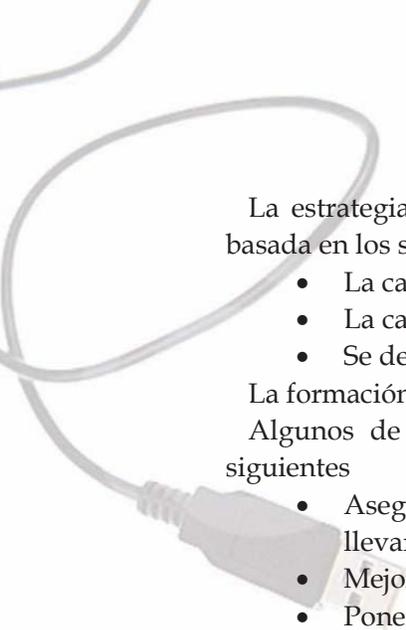
Hasta hace poco tiempo que la calidad del software no se le prestaba atención debido a las fuertes dificultades:

- Incomprensión de las necesidades del usuario del producto.
- La inexistencia de requisitos de calidad en las especificaciones.
- La necesidad de llegar a las personas correctas para realizar los diversos trabajos.
- La necesidad de formar a los Formadores en calidad.
- La necesidad de ayuda en lugar de control
- La necesidad de comprender las consideraciones del negocio.

Los programas de calidad del software son instrumentos utilizados por la dirección para asegurar que se han tomado las medidas adecuadas para construir la calidad en el software.

Estos programas deben contemplar todos los elementos de la calidad del software. Estos elementos son:

- Educación y formación del personal.
- Definición del proceso de desarrollo.
- Asegurar el cumplimiento al proceso decidido.
- Adquisición o desarrollo de herramientas.
- Eliminación de defectos, y
- Medidas y análisis.



La estrategia de la introducción de la función de la calidad del software (FCASW) estará basada en los siguientes aspectos:

- La calidad debe percibirse con resultados tangibles inmediatos.
- La calidad deberá venderse y hacerlo como una función que genera beneficios.
- Se deberá establecer credibilidad del grupo de calidad respecto a su competencia.

La formación y el desarrollo profesional del personal es un proceso que nunca acaba.

Algunos de los principales objetivos de la formación en calidad del software son los siguientes

- Asegurar que las personas implicadas en la calidad del software están en posición de llevar a cabo sus tareas de una Forma afectiva y eficiente.
- Mejorar la conciencia en el personal asociado a un proyecto dado.
- Poner el énfasis en aquellas áreas de actividad consideradas a las que haya que aplicarse un esfuerzo de Mejora de la calidad.

Los siguientes aspectos son de interés respecto a la formación:

1. Identificación de áreas débiles
2. Previsión de las necesidades del personal.
3. Planificación por adelantado para cumplir las necesidades futuras de la organización.
4. Hacer presentaciones sobre los Últimos avances en calidad del software.

Otra parte define las métricas de calidad que han de cumplirse por los procesos y productos acabados. La calidad de los diseños y desarrollos representan el saber hacer de la empresa.

LA POLITICA PROFESIONAL.

El desarrollo de software de calidad requiere de buenos diseñadores, buen personal de calidad y de buenos gestores. La demanda de los mismos supera a la oferta en la actualidad y en los próximos años el desequilibrio aún será mayor.

Hay que considerar esenciales los siguientes aspectos en relación a la gestión de grupos implicados en el desarrollo software:

- La dirección de creación debe comprender el proceso total de creación del software y debe ser capaz de comunicarse en el mismo lenguaje que el de los diseñadores y personal de calidad.
- Desde el punto de vista de la gestión de personal se necesitan las siguientes etapas:
 - Evaluación del personal e identificación de los más expertos.
 - Establecimiento de planes de desarrollo profesional y vías de promoción. Esto incluye aprendizaje con el personal más experto, cursos y educación formal.
- Asignación de los escasos recursos humanos de alto nivel a los proyectos más complejos.

La dirección de organización debe estar muy estrechamente relacionada al entorno del desarrollo software, por lo que deberá incluir:

- Planes y presupuestos adecuados para los proyectos,
- Control del progreso,
- Procedimiento de pruebas de aceptación, modificaciones, mejoras, etc, bien definidas.
- Control formal de cambios de la documentación.

MODELO CONCEPTUAL DE UN SISTEMA DE FORMACIÓN PROFESIONAL.

El modelo que se presenta aquí puede definirse como un sistema para diseñar y gestionar sistemas para seleccionar, formar y evaluar al personal de una organización dedicado a actividades relacionadas con la calidad del software, desde el punto de vista del desarrollo, la

gestión de la Calidad o las tecnologías aplicadas. Tal modelo no es un diagrama organizativo. Describe una estructura de relaciones entre las funciones de diseño de sistemas de formación, diseño de sistemas de evaluación, proceso de datos y otros sistemas. En el modelo, fig. 1- hay tres niveles de jerarquía de los diversos sistemas que constituyen el modelo de sistema de recursos humanos.

En cada nivel cada uno de los sistemas debe cumplir una función específica, y a su vez están compuestos por subsistemas que son sistemas funcionales. El sistema de Diseño de Formación es un sistema que puede ser una persona, dos, un grupo o un sistema CAD.

Estas interacciones son funciones compartidas, intercambios entre sistemas.

Así la identificación de los requisitos de selección, puede actuar como entrada al sistema de formación, describiendo como dicho sistema interactúa con el sistema de selección. Cada función puede refinarse por medio de nuevas especificaciones que describan las interacciones de cada sistema.

El Sistema de Diseño de la Formación puede definirse por las siguientes especificaciones:

- Descripción de las posibilidades u oportunidades de formación.
- Especificación de los requisitos de formación, realizada junto con la dirección.
- Especificación de los objetivos de formación.
- Diseño del sistema de formalización para realizar estos objetivos.
- Diseño de los programas de formación para estos sistemas.

La fig. 2 es una expansión del Sistema de la formación y representa el modelo conceptual de un Sistema de Formación.

En este modelo se identifica al cliente del Servicio de Formación que puede ser la alta dirección o el Jefe de Desarrollo de Sistemas. Así si la alta dirección tiene un programa de gestión de la calidad del software para directivos medios, el Sistema de Formación desarrollará aquéllos programas que den servicio a los objetivos de la alta dirección. En las cajas se identifican los componentes de la organización que realizan funciones de formación (diseño de programas, sistema de desarrollo, sistema de evaluación, sistema de producción de los programas).

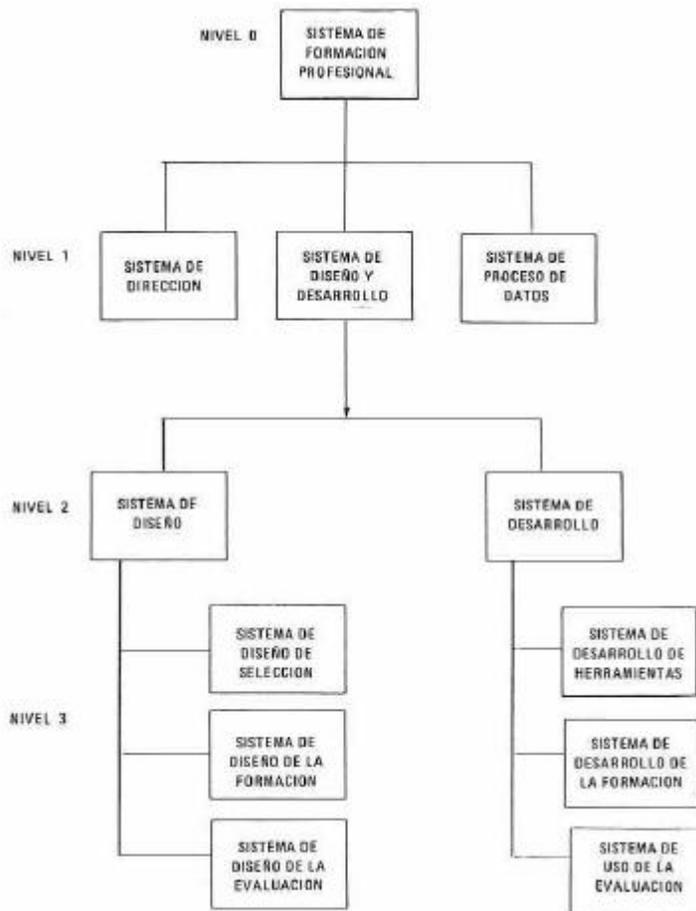
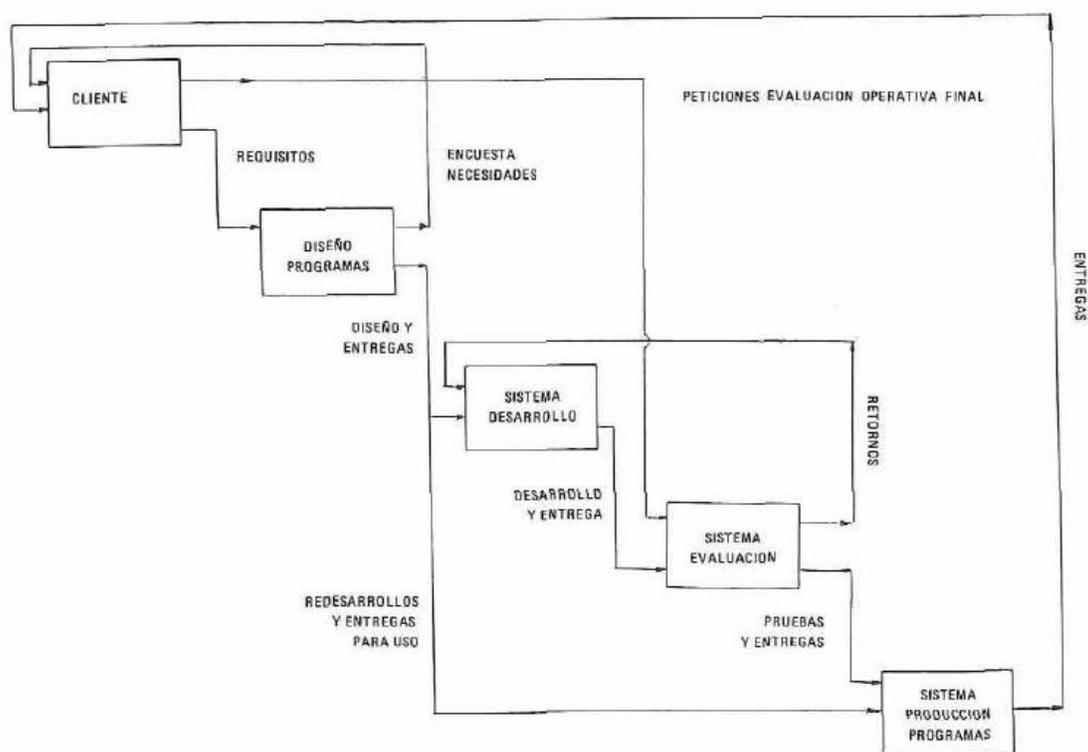


Fig. 2.- MODELO CONCEPTUAL DE UN SISTEMA DE FORMACION



El flujo de actividades está representado por flechas que indican hacia donde se dirigen las actividades y palabras que indican la naturaleza de las mismas.

En este modelo, el cliente solicita formación; el grupo de diseño de programas encuesta las necesidades del cliente y éste describe sus requisitos: El sistema de diseño de programas, entonces, diseña un Sistema de Formación y lo entrega al Sistema de Desarrollo, que realiza un Sistema de Formación Operativo, que a su vez es remitido al Sistema de Evaluación para prueba. Si la evaluación es negativa, el sistema de formación se devuelve a Desarrollo y se repiten las etapas. Si es positiva, el programa se entrega al Sistema de Producción de Programas que entrega e imparte el producto al cliente. Finalmente el Programa de Formación a funcionar se evalúa cuando se entrega.

Este modelo se puede refinar. El Sistema de Diseño de Programas se puede dividir en varios subsistemas, p. ej., Sistema de Formulación de Objetivos, Sistema de Modelado, etc.

El modelo conceptual sirve para identificar las funciones necesarias para producir un resultado, alcanzar un objetivo y hacer un producto de formación, así como establecer las interrelaciones de control de dichas funciones.

LAS DISCIPLINAS Y PROGRAMAS DE FORMACIÓN EN CALIDAD DEL SOFTWARE.

Las disciplinas de la calidad del software son un conjunto de conceptos, métodos, herramientas, técnicas y competencias relativas a la calidad del software, gracias a las cuales gestionamos la función de calidad del software.

Algunos componentes de las disciplinas de la calidad del software se han desarrollado en otras funciones, en particular en la ingeniería software y la gestión general de la Calidad. Otros componentes se han desarrollado en forma específica para dar soporte a la función de Calidad del Software.

El ámbito del programa de formación debe alcanzar tanto desde los niveles de Dirección de Calidad y de Ingeniería de Diseño y Desarrollo, pasando por jefes de proyectos, personal de especificación, diseño, desarrollo pruebas, integración, aceptación, homologación y mantenimiento, así como el de auditoría, el de gestión y garantía de calidad, el de verificación y validación y el de mantenimiento.

LA GESTION DE LA CALIDAD DEL SOFTWARE (GCSW).

Parafraseando el concepto de Gestión de la Calidad de ISO 8402, hemos definido la GCSW como "el aspecto de la función general de la gestión que determina y aplica la política de la calidad del Software".

El papel de la GCSW es el comprobar que los procedimientos de calidad acordados contractualmente se han planificado y están implantados los grupos de desarrollo.

Las actividades de GCSW deben estar estrechamente coordinadas con la Dirección de Desarrollo. Dichas actividades son:

- Producción de un plan de calidad de acuerdo con las normas acordadas en contrato.
- Las auditorías del control de configuración para los componentes del producto.
- Las auditorías del control de cambios para los documentos de diseño del producto.
- La identificación de los requisitos de calidad para un producto.

El esquema de la –fig. 3– muestra el contexto.



calidad del software son:

- El control de defectos.
- La identificación y documentación de los procedimientos.
- Las auditorías y revisiones de los proyectos.
- El establecimiento de los costes de calidad.
- La impartición de formación en los procedimientos y métodos del aseguramiento de Calidad.

Las revisiones e inspecciones de los requisitos de A.C. SW del suministrador, deberán tener como propósitos principales:

- Comprobar si existe suficiente guía por parte de la normativa para la realización de tales evaluaciones.
- Permitir el examen de los procedimientos de los suministradores y determinar si son adecuados para minimizar los errores potenciales no detectados que tienen impacto en el software.

El software asociado y desarrollado significa aquél software especial necesario para soporte de distintas actividades en cada una de las fases del ciclo de vida software.

El software comercial no está en esta categoría. Sólo los productos de trabajo desarrollados

- La planificación de la inspección y procedimientos de prueba conformes a las normas de calidad acordadas en el contrato.

- El aseguramiento de que todos los trabajos de calidad planificados tienen lugar con el personal apropiado y en los tiempos previstos.

- El establecimiento de etapas para asegurar la prevención, detección, eliminación y análisis de defectos.

Aspectos esenciales del aseguramiento de la

en paralelo al producto contratado son los objetos a tener en cuenta a entregar, y de la calidad del mismo va a depender la calidad del producto del cliente.

Dentro del plan de calidad de un proyecto de desarrollo el suministrador deberá indicar los requisitos de calidad a sus propios suministradores en relación al:

- Software a desarrollar específicamente por los subcontratistas.
- Software comprado, alquilado o impuesto.

Se deben describir o referenciar los procedimientos de adquisición de los productos realizados fuera de la responsabilidad del suministrador teniendo en cuenta los siguientes aspectos:

- La evaluación del producto en relación a las necesidades.
- Condiciones de aceptación y de garantía.

Un programa mínimo de G.C.SW deberá contener las siguientes materias:

- Conceptos y definiciones: calidad, política, sistema control, aseguramiento, revisiones y auditoría de la calidad del software.
- Objetivos y principios.
- La función de la Calidad del Software.
- La planificación del producto y la gestión y control del proyecto.
- Los sistemas de Calidad: ISO, AQAP, otros.
- El Manual de Calidad.
- Los planes (de Calidad, de Gestión, de Configuración, de Verificación y Validación).
- Los procedimientos.
- Las normas.
- Los Programas de Mejora.
- Las Técnicas de Control.
- Las Revisiones y Auditorías.
- La puesta en marcha de un Plan de Calidad.
- Las actividades de Metrología
- Los Costes de la Calidad.
- Los aspectos contractuales de la Calidad del Software.

Esta formación deberá impartirse a todos aquellos que estén implicados en el área de aseguramiento y control de calidad, pero también a la dirección de las organizaciones y a los grupos de diseño y desarrollo a los responsables de la adquisición y contratación de los productos de software.

CONCLUSIONES.

La implantación de un servicio de formación en ingeniería y calidad del software no es tarea fácil. En primer lugar hay que justificar como elemento clave de competitividad. No se hacen productos de calidad sin una base amplia y específica de conocimientos actualizados en los aspectos de ingeniería, de gestión y de calidad que intervienen en el complejo proceso de creación del software.

La introducción en la empresa de una formación adecuada pasa por disponer de algún modelo que permita organizarla para ser efectiva. En la práctica se muestra poco efectiva las soluciones típicas de aprovechar algunos cursos dispersos sobre temas particulares, ya que no proporcionan todos los elementos necesarios para abordar la formación necesaria, e integrada, requerida por la industria software.

REFERENCIAS

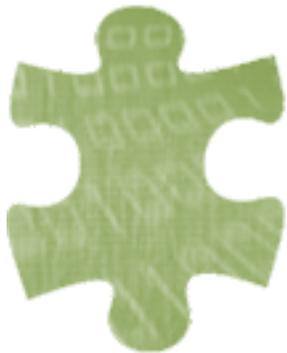
- BRAVO C. y MACHO ORTIZ, J.I. *La Gestión de la Configuración del Software Seminario de la A.E.C.C., Madrid, 1988.*
- Com. *Calidad del Software Glosario de Términos de Calidad e Ing. del Software A.E.C.C., Madrid, 1986.*
- COM. CALIDAD DEL SOFTWARE *Guía para la Realización de Planes de Garantía de Calidad del Software. A.E.C.C., Madrid, 1987.*
- COPER, J.D. y FISHER, MJ. *Software Quality Management. PBI, 1979.*
- EGERMAYER, F. *Global Quality Education and Training Program. Rocrd. World Quality Congress 84 EOQCIIQAJ /BQA, Vol. 1, pp. 477-487, Brighton, 1984.*
- GONZALEZ SANZ, J. *La garantía de la Calidad de los productos Software hoy. Convención Iberoamericana de Informática (CIBI-85). Madrid, 1985.*
- GONZALEZ SANZ, J. *La Calidad del Software Conf. del curso: "La Gestión de la Calidad en la Industria Electrónica" Univ. Politécnica de Madrid - A.E.C.C. - Madrid, mayo 1985.*
- GONZALEZ SANZ J. y MACHO ORTIZ, J.I. *La Gestión de la Calidad del Software Seminario de la A.E.C.C., Madrid, 1985.*
- GONZALEZ SANZ, J. y BRAVO ORTIZ, C. *La normalización en Ingeniería y Calidad del Software. Seminario de la A.E.C.C., Madrid, 1987.*
- GONZALEZ SANZ, J. y GARCIA HIGUERA, La *Fiabilidad del Software. Seminario de la A.E.C.C., Madrid, 1988.*
- GONZALEZ SANZ, J. y RODRIGUEZ, P. *Las Métricas de Calidad del Software. Seminario de la A.E.C.C., Madrid, 1989 (en preparación).*
- GONZALEZ SANZ, J *La Calidad del Software en los Sistemas de Comunicaciones de Datos. Feria TECNOTRON-LIMA Junio, 1985.*
- GONZALEZ SANZ, J. *La Qualité et L'Assurance de la Qualité dans les Industries du Logiciel Anjourd'hui. 1 Congres en France sur "La Génie Industriel Facteur de Competitivité des Entrepsdes". AFCET/CEFI/GGI/. Paris, Jnne 1986.*
- GONZALEZ SANZ, J. *Panorama Actuel des Activités de Normabarion en Gbnie et Qualité du Logiciel. 2^e Sem. Europeen Com. Technique de Normalisation en Gestion de la Qualité du EOQC., Paris, septiembere 1986.*
- GONZALEZ SANZ, J. *Ingeniería Software. Curso del Master de Informática Avanzada MiNER celebrado en la EOI., Madrid, 1987.*
- GONZALEZ SANZ, J. *La garantía de Calidad del Software.*
- GONZALEZ SANZ, J. *Standards for Quality and Quality Assurance in Software Industries Today. 31 Conf. EOQC., Munich, 1987.*
- GONZALEZ SANZ, J. *Software Quality Assurance in a Large Data Packets Switching System. 32 Conf. EOQC., Moscow, 1988.*
- GONZALEZ SANZ, J. *Defining Tems relating to software Quality and Reliability Round Table Dis. on: communicating Information New Requirements for Terminology. Com. Terminología de la EOQC., Moscú, 1988.*
- GONZALEZ SANZ, J. *La Calidad Industrial en el umbral de los 90. Bol. CALIDAD de la AECC, no 2. Marzo-Abril. Moscow, 1988.*
- GONZALEZ SANZ, J. *La Calidad y la Fiabilidad del Software. Conf. del curso: "La Gestión de la Calidad en la Industria Electrónica". Esc. Técnica Sup. Ing. Telecomunicaciones y Asoc. Nac. Ind. Electrónicas (MIEL). Madrid, 1987 /Bilbao, 1988.*
- GONZALEZ SANZ, J. *Ingeniería y Gestión del Desarrollo Software. Seminario CONFEMETAL, Junio 1988.*
- GONZALEZ SANZ, J. *Ingeniería y Calidad del Software. Seminario OMNITEL, Mayo 1988.*
- GONZALEZ SANZ, J. *The importance of Q.A. Management for no deliverable software. V SYMP. NATO: NATO QUALITY ASSURANCE FOR THE YEAR 2000 (a celebrar en Madrid el 1 Junio 1989).*
- GROOCOOCK, J M. *The TRW Quality Cdege. Procd. 27 th EOQC. Conference, pp. B6-211 B6-219, Madrid, 1983.*
- JURAN, J.M. *Management of Quality. McGraw Hill BookCo. 4a ed., a981. 27.- LYSY, K.A. Software Quality Engineering and Structured Methody. Procd. COMPSAC-87, pp. 103-109,1987.*
- MUNERA, J y GONZALEZ SANZ, J. *Software*

Reliability. Procd. sobre "Advanced Systems reliability modeling". ISPRA Courses. Madrid 19-23 sept. 1988.

UNE 66-001-88 Calidad- Vocabulario. (Doc. Trabajo). AENOR, Madrid, Febrero 1988. I



personas

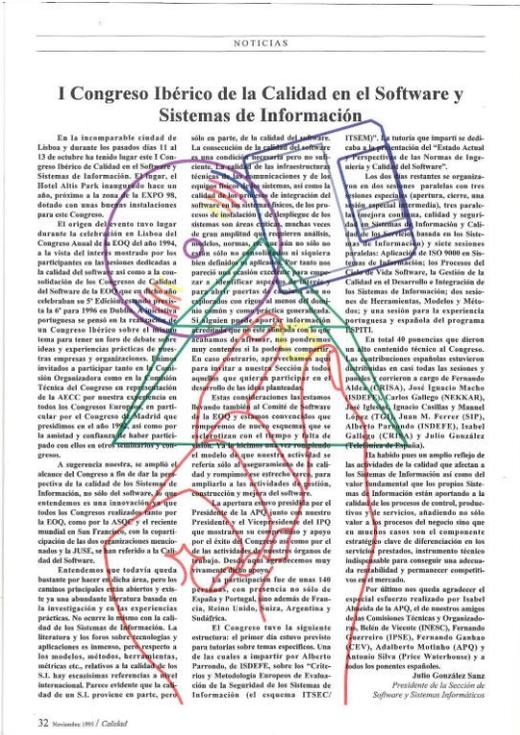


procesos



tecnologías

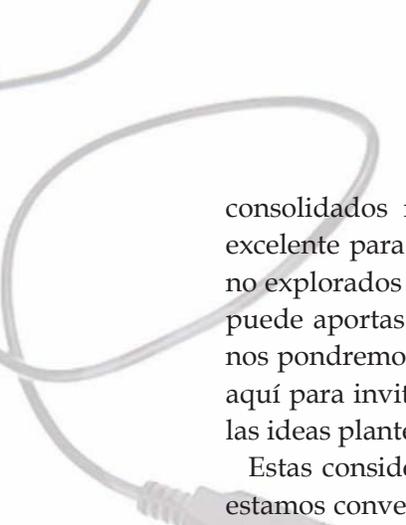
I Congreso Ibérico de la Calidad en el Software y Sistemas de Información



En la incomparable ciudad de Lisboa y durante los pasados días 11 al 13 de octubre ha tenido lugar este, I Congreso Ibérico de la Calidad en el Software y Sistemas de Información. El lugar, el Hotel Altis Park inaugurado hace un año, próximo a la zona de la EXPO98, dotado con unas buenas instalaciones para este Congreso. El origen del evento tuvo lugar durante la celebración en Lisboa del Congreso Anual de la EOQ del año 1994, a la vista del interés mostrado por los participantes en las sesiones dedicadas a la calidad del software así como a la consolidación de los Congresos de Calidad del Software de la EOQ que en dicho año celebraban su 5ª Edición estando prevista la 6ª para 1996 en Dublín A iniciativa portuguesa se pensó en la realización de un congreso Ibérico sobre el mismo tema para tener un foro de debate sobre ideas y experiencias prácticas de nuestras empresas y organizaciones. Como invitado a participar tanto en la Comisión Organizadora como en la Comisión Técnica del Congreso en representación de la AECC por nuestra experiencia en todos los Congresos Europeos en particular por el Congreso globalizado que presidimos en el año 1992, así como por la amistad y confianza de haber participado con ellos en otros seminarios y congresos.

A sugerencia nuestra, se amplió el alcance del Congreso a fin de dar la perspectiva de la calidad de los Sistemas de Información, no sólo del software, lo que entendemos es una innovación ya que todos los Congresos realizados tanto por la EOQ, como por la ASQC y el reciente mundial en San Francisco, con la coparticipación de las dos organizaciones mencionadas y la JUSE, se han referido a la Calidad del Software.

Entendemos que todavía queda bastante por hacer en dicha área, pero los caminos principales están abiertos y existe ya una abundante literatura basada en la investigación y en las experiencias prácticas. No ocurre lo mismo con la calidad de los Sistemas de Información. La literatura y los foros sobre tecnologías y aplicaciones es inmenso, pero respecto a los métodos, modelos, herramientas, métricas etc., relativos a la calidad de los S.I. hay escasísimas referencias a nivel internacional. Parece evidente que la calidad de un S.I. proviene en parte, pero sólo en parte de la Calidad del Software. La consecución de la calidad del Software es una condición necesaria pero no suficiente. La calidad de las infraestructuras técnicas de las comunicaciones y de los equipos físicos de los sistemas, así como la calidad de los procesos de integración del software en los sistemas físicos, de los procesos de instalación y despliegue de los sistemas son áreas críticas, muchas veces de gran amplitud que requieren análisis, modelos, normas, etc., que aún no sólo no están sólo no



consolidados ni siquiera bien definidos ni aplicados. Por tanto nos pareció una ocasión excelente para empezar a identificar aspectos de interés y para abrir puertas de caminos aún no explorados con rigor al menos del dominio común y como práctica generalizada. Si alguien puede aportar información acreditada que no esté alineada con lo que acabamos de afirmar, nos pondremos muy contentos si la podemos compartir. En el caso contrario, aprovecharemos aquí para invitar a nuestra sección a todos aquellos que quieran participar en el desarrollo de las ideas planteadas.

Estas consideraciones las estamos llevando también en el Comité de Software de la EOQ y estamos convencidos que romperemos de nuevo esquemas que se esclerotizan con el tiempo y falta de visión. Ya lo hicimos una vez rompiendo el modelo de nuestra actividad, se refería sólo al aseguramiento de la calidad y rompimos este estrecho cerco, para ampliarlo a las actividades de gestión, construcción y mejora del software.

La apertura estuvo presidida por el presidente de la APQ junto con nuestro presidente y el vicepresidente del IPQ que mostraron su compromiso y apoyo por el éxito del Congreso así como por el de las actividades de nuestros órganos de trabajo. Desde aquí agradecemos muy vivamente dicho apoyo.

La participación fue de unas 140 personas, con presencia no sólo de España y Portugal, sino además de Francia, Reino Unido, Suiza, Argentina, y Sudáfrica.

Ha habido pues un amplio reflejo de las actividades de la calidad que afectan a los Sistemas de Información así como del valor fundamental que los propios Sistemas de Información están aportando a la calidad de los procesos de control, productivos y de servicios, añadiendo no sólo valor a los procesos del negocio sino que en muchos casos son el componente estratégico clave de diferenciación en los servicios prestados, instrumento técnico indispensable para conseguir una adecuada rentabilidad y permanecer competitivos en el mercado.

Por último nos queda agradecer el especial esfuerzo realizado por Isabel Almeida de la APQ, el de nuestros amigos de las Comisiones Técnicas y Organizadoras, Belén de Vicente (INESC), Fernando Guerreiro (IPSE), Fernando Ganhao (CEV), Adalberto Motiuho (APQ) y Antonio Silva (Pricce Waterhouse) y a todos los ponentes españoles.

Julio González Sanz

Las empresas de Tecnologías de la información y la nueva economía



Autor: Eduardo Vilalta Pallares Director de Marketing INDRA

Publicado en la revista Calidad de Junio de 2001

El presente artículo está basado en una de las ponencias presentadas en la I Cumbre de Excelencia en la Gestión, celebrada en Madrid los días 21 y 22 de junio del pasado año. El impacto de internet en la actividad económica ha provocado la necesidad de adaptar los modelos de negocio a las empresas. Para asistirles en estos procesos, las compañías de servicios de tecnologías de la información deben evolucionar en su perfil y en su oferta para responder a los nuevos requerimientos.

La imparable globalización de los mercados caracterizada, básicamente, por supresiones de barreras geográficas, monetarias, comerciales, etc., está dando lugar a un profundo cambio, de consecuencias desconocidas, tanto en la sociedad como en las relaciones entre los individuos.

Uno de los grandes catalizadores de este cambio es la innovación tecnológica, y, muy especialmente, la que se está desarrollando alrededor de Internet sobre terminales fijos y móviles, iTV, etc., que está permitiendo generar las condiciones adecuadas para el desarrollo de este fenómeno.

Este nuevo entorno impacta de forma decisiva en las formas tradicionales de hacer negocio y de interactuar las organizaciones, lo que conduce, necesariamente, a definir la visión de un nuevo paradigma, la economía digital, en la que el elemento diferenciador no es el "tamaño" de las organizaciones como venía siendo habitual, sino su "rapidez" de actuación frente a los cambios.

En este sentido, nunca como en estos momentos, se habían presentado tantos retos simultáneos como los planteados por la economía basada en Internet. Factores como la gestión del conocimiento, la capacidad de innovación, la ubicuidad en el espacio. La reducción de los tiempos, las alianzas y nuevos canales, el acortamiento de los ciclos de vida y, por encima de todos ellos, la focalización en torno al CLIENTE, son los ejes directores de las nuevas organizaciones digitales.

Por otro lado, no basta con disponer de esta visión de futuro; los líderes de esta nueva economía digital deben poseer, además, la capacidad de ejecutarla rápida, simultánea y globalmente, alineando, de forma permanente, sus estrategias de negocio con este entorno en turbulencia continua.

Este escenario de cambio Permanente obliga a las organizaciones, desde un punto de vista

interno, a adelantarse a la competencia y a las exigencias crecientes del mercado, mediante un esfuerzo permanente de agregación de valor y reducción de costes.

Igualmente, y desde un punto de vista externo, estas organizaciones digitales deberán aprovechar al máximo las nuevas oportunidades que surgen como consecuencia de la creación de nuevos "espacios digitales de negocio" y de la gestión óptima de los riesgos de participación en nuevas iniciativas.

LOS INTEGRADORES INTERACTIVOS

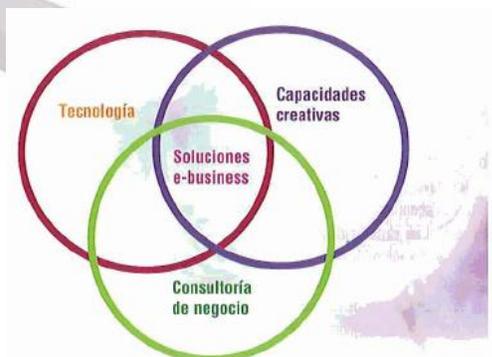
En la primera etapa de utilización de Internet en nuestro entorno económico, la Red era percibida como un mero canal adicional de comunicación con clientes potenciales. En consecuencia la demanda inicial de servicios a suministrar por parte de empresas especializadas (Agentes Interactivos) vino determinada por lo que entendemos como la parte Business to Consumer (B2C) de la revolución que ha supuesto la economía digital. Los contratos para este tipo de servicios han sido de un tamaño relativamente pequeño y han requerido una elevada proporción de capacidades front end de diseño de web, marketing y habilidades multimedia. Los demandantes han necesitado ayuda en el diseño de web sites que pudieran captar y retener la atención de sus posibles clientes como factor clave del éxito en el mundo del comercio electrónico. Las capacidades requeridas a los proveedores de servicios en esta etapa no han tenido demasiado que ver, en realidad, con las Tecnologías de la Información y la mayoría de las compañías tradicionales del sector, no se han involucrado mucho en esta actividad.

Sin embargo, la Red ha evolucionado como concepto y hoy en día se considera un medio para que las empresas puedan gestionar de forma optimizada sus cadenas de suministro, sus relaciones con los clientes y sus principales activos intangibles (los conocimientos de sus empleados y colaboradores). Es decir, la red se ha convertido en un "facilitador esencial" de la actividad económica. La mayoría de las empresas en cualquier sector se ven en la necesidad de hacer evolucionar sus modelos de negocio para adaptarse a este entorno cambiante y, sobre todo a sus exigencias de tiempos de reacción muy cortos (time to market). Los proyectos de Internet deben alinearse con la estrategia de negocio y con los procesos de las organizaciones. Por lo tanto, dichos proyectos pasan a estar íntimamente relacionados con la estrategia global de la empresa de la que constituyen una parte importante sus Sistemas de Tecnologías de la Información, con los que los proyectos de Internet de la empresa deben estar totalmente integrados. En consecuencia, para responder a la demanda de servicios que las empresas - cliente exigen a los suministradores, en su adaptación a la economía digital, hace falta un perfil distinto al de las compañías de servicios de Tecnologías de la información tradicionales, puesto que su oferta debe incluir una combinación de consultoría estratégica, capacidades creativas y tecnologías de integración de software. Este perfil es el que podemos denominar como "Integrador Interactivo". La palabra interactivo se refiere no solamente a la naturaleza de los web sites que estas compañías diseñan, sino también a un proceso de negocio que requiere realimentación constante por parte del cliente. La denominación de Integradores refleja la importancia creciente de la interfase entre el front-end de la web del cliente y los sistemas de Tecnologías de la Información de este último. Debido a que la solución que estas compañías proporcionan es fundamental para los negocios de sus clientes, su oferta se dirige a la alta dirección de los mismos, en contraste con los Agentes Interactivos, que acostumbran a dirigirla a los departamentos de marketing y con los integradores de sistemas tradicionales, cuyo punto

de contacto es normalmente el departamento de Tecnologías de la Información

REQUERIMIENTOS DEL MERCADO

Sin pretender ser exhaustivos, vamos a tratar de entrar con algún nivel de detalle en las características de los servicios demandados por los clientes y a los que deben responder los Integradores Interactivos.



Consultoría estratégica en Internet

La consultoría estratégica en Internet constituye una de las "joyas de la corona" para los suministradores de servicios en la Red. El Liderazgo en este campo de actividad está asociado a una profundización en las capacidades de consultoría de negocio y estrategia. Al dirigirse las organizaciones a la Red en busca de nuevas oportunidades de negocio, requerirán sin duda la asistencia de consultores en

distintas áreas de competencia, incluyendo:

- Análisis, evolución e impacto de la tecnología y aplicaciones de Internet en los negocios digitales
- Puesta en marcha y participación en nuevas iniciativas empresariales digitales
- Redefinición estratégica de negocios
- Análisis de viabilidad de nuevos modelos de negocio
- Rediseños organizativos y gestión del cambio

Es interesante notar que aunque las compañías "puras" de Internet, es decir, que han surgido como especializadas y enfocadas sólo a la Red, son percibidas como los auténticos "expertos" en Internet. Bastantes de ellas carecen de la amplitud y profundidad de capacidades para ayudar a las organizaciones en su evolución a la nueva economía, sobre todo en lo que se refiere a estrategia de negocio, rediseño organizativo y gestión del cambio. También hay que destacar el hecho de que las grandes consultoras se han dado cuenta de la importancia de disponer de capacidades en este entorno y están captando recursos en cantidades apreciables.

Capacidades creativas

En realidad se trata de elementos de marketing/branding enfocados a la Red. Desde el punto de vista de marketing implica el análisis, detección y fidelización de los nuevos clientes a través de los canales digitales, que comprende las siguientes líneas de actividad:

- Diseño de portales horizontales y verticales
- Diseño e implementación de nuevas estrategias de marketing para negocios en Internet
- Evaluación del nivel de satisfacción y fidelización de clientes
- Análisis y desarrollo de estrategias de marketing one to one

Igualmente, desde la óptica del branding, implica el análisis y lanzamiento promocional de nuevas imágenes de marca en la Red, e incluye, a su vez, las siguientes líneas:

- Diseño, construcción y lanzamiento de marcas
- Desarrollo de la marca e integración con las estrategias corporativas

Tecnologías de integración

Se trata de la capacidad de ofrecer soluciones y servicios en áreas como:

- Comercio electrónico
- Customer Relationship Management (CRM)
- Supply Chain Management (SCM)
- Seguridad en la Red

El comercio electrónico

Engloba los conceptos tradicionales de intercambio de información y pagos entre empresas (B2B), tales como:

- Procesos de aprovisionamiento
- Gestión de subastas
- Presentación electrónica de facturas
- Operaciones logísticas
- Centros de mercado (market places)

Así mismo se pueden añadir los intercambios y pagos seguros entre empresas y consumidores finales (B2C), entre los que pueden mencionarse:

- Diseño y gestión de tiendas y "malls" virtuales
- Gestión de contenidos en red
- Portales de negocio horizontales (servicio de correo, canales temáticos, servicios de publicación, etc.)
- Portales verticales (consulta de movimientos bancarios, atención a clientes, informaciones de productos, etc.)
- Brokerage financiero (contrataciones en Bolsa, mercados de valores, títulos y cuentas bancarias, etc.)

El Customer Relationship Management

Permite a las organizaciones gestionar óptimamente sus relaciones con los clientes, dirigiendo sus actividades hacia una atención personalizada, una mejor comprensión de sus necesidades y expectativas y una mayor fidelización, con objeto de conseguir la máxima rentabilidad en la relación.

Ello implica soluciones y servicios como:

- Contact Centers integrados con portales en la Red
- Help desk tanto para gestión interna como para seguimiento de clientes
- Dispatching de suministros, rutas y recursos
- Tarifación por canales, productos y mercados
- Control y seguimiento de las actividades comerciales
- Automatización de la fuerza de ventas, incluyendo la situación del cliente, control de stocks, gestión de la agenda de contactos y visitas, etc.
- Análisis de ventas para obtención de patrones, rentabilidad, efectividad de los canales, etc.
- Definición de políticas y mecanismos de aplicación de incentivos
- Análisis y segmentaciones de mercados y clientes
- Diseño de campañas comerciales y de marketing

- Modelos predictivos de mercados y demandas potenciales
- Análisis de necesidades, hábitos y rentabilidad
- Balanced scorecard para la valoración del impacto de las decisiones y las estrategias.

El Supply Chain Management (SCM)

Con su visión más generalizada de "planificación colaborativa" que facilita y optimiza la gestión de los suministros, dentro de cada uno de los elementos que componen la cadena logística de una organización, integrándolos tanto desde el punto de vista de sus clientes como de sus proveedores, lo que ya se empieza a conocer en el mercado como la "empresa ampliada".

El S.C.M. incluye, entre otros, soluciones y servicios como:

- e-procurement, gestión integral de aprovisionamientos, que incluye aspectos como la dinamización de los procesos corporativos de compra
- Generación de catálogos on-line de productos
- Gestión de portales y centros de comercio para la realización de transacciones de intercambio y subastas
- Gestión integrada de inventarios, almacenes, y toda la cadena logística
- Integración con E.R.P.'s y workflows
- Planificaciones colaborativas

La seguridad en la Red

Se focaliza en garantizar la seguridad en las diferentes interacciones con las redes mediante la identificación y autenticación, el control de accesos, el no repudio y su máxima disponibilidad. Para ello, hay que tener en cuenta algunos elementos clave como:

- Registro y gestión de certificados públicos (PKI's)
- Uso de tarjetas inteligentes
- Sellado de tiempo
- Definición de perfiles de acceso

Igualmente, es preciso garantizar no sólo la absoluta seguridad en las transacciones mediante el empleo de:

- Firmas digitales y encriptación
- Correo y webs securizadas
- Certificados EDi
- Middlewares cifrados y firmados

Sino también en las redes e infraestructuras, a través de.

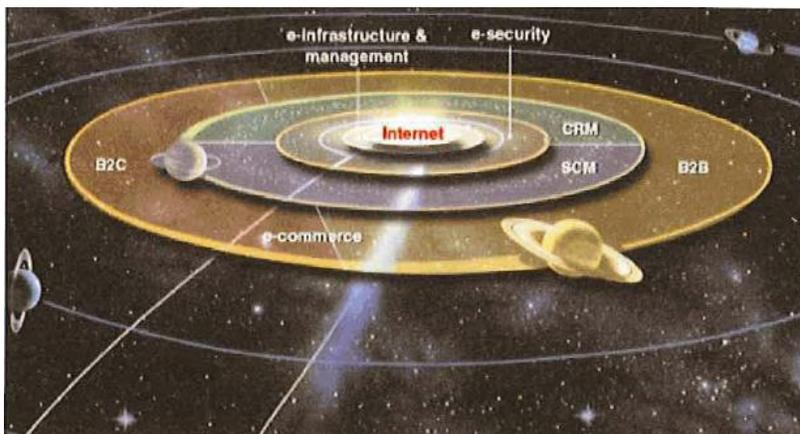
- Securitización de conexiones entre nodos de redes virtuales en Internet
- Conexiones de accesos remotos seguros
- Implantación de sistemas de detección de virus
- Firewalls
- Autenticaciones biométricas

CUESTIONES CLAVE PARA UN INTEGRADOR INTERACTIVO

Hay una serie de elementos fundamentales a tener en cuenta para obtener ventajas competitivas en el mercado de los servicios de Red.

Énfasis en la atracción y retención de empleados

La capacidad para dotar a los proyectos de Internet de suficiente personal capacitado constituye un factor crucial. Para ello es necesario poner en marcha programas de alto nivel para reclutamiento y retención de profesionales. No solamente hacen falta paquetes de compensación adecuados e imaginativos, sino también hay que saber proporcionar beneficios



menos tangibles (trabajar con tecnología punta, participar en proyectos variados e interesantes, flexibilidad en horarios de trabajo, oportunidades de desarrollo de carreras, formación continuada y un sentido de "propiedad" de la compañía).

Para una organización que trate de combinar culturas creativas y técnicas es

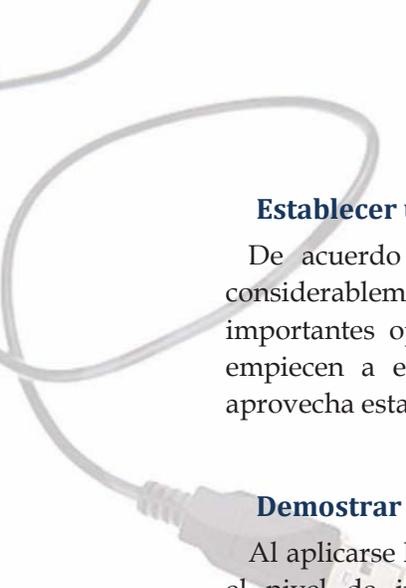
importante desarrollar una cultura "de síntesis" que promueva los méritos de los dos tipos de capacidades y aglutine a los correspondientes tipos de empleados alrededor de una visión común.

Desarrollo de competencias en Gestión del Cambio

Los proyectos de Internet implican con frecuencia cambios significativos en la forma en que una empresa lleva a cabo sus actividades. Debido a que dichos proyectos afectan a diversas áreas dentro de una organización (o entre organizaciones), es importante para el suministrador de servicios actuar como intermediario. A pesar de los problemas internos que puedan tener lugar en los clientes la responsabilidad de entregar una solución completa a tiempo, recae en el suministrador de servicios. Los suministradores que se han preocupado de desarrollar u obtener competencias de Gestión del Cambio, estarán en mejor posición para llevar a cabo este tipo de proyectos.

Suministro de soluciones multidisciplinarias completas

La demanda de soluciones Internet completas (soluciones que incluyen estrategia de negocio, marketing, tecnologías y servicios) será cada vez más importante. Los suministradores de servicios que tengan carencias en alguna de estas capacidades, tendrán que desarrollar o adquirir las correspondientes competencias.



Establecer una presencia global

De acuerdo con estudios del sector, el mercado de los servicios de internet crecerá considerablemente durante los próximos cinco años (del orden del 60-70%), y habrá importantes oportunidades de negocio. Los suministradores de este tipo de servicios que empiecen a establecer su presencia internacional ahora, estarán bien posicionados para aprovechar estas oportunidades.

Demostrar el retorno de la inversión en proyectos.

Al aplicarse los gestores de línea en los procesos de compra y toma de decisiones, y al crecer el nivel de inversión en los proyectos de internet, resulta particularmente importante demostrar el retorno de la inversión que los proyectos de la Red puedan proporcionar. En consecuencia, habrá que desarrollar metodologías y herramientas para medir ROI, teniendo en cuenta que hay beneficios en proyectos de internet que no son siempre tangibles (más participación de mercado o mejor imagen de marca).

Por último es fundamental transmitir a los clientes el compromiso con los servicios de Internet y la confianza en las propias competencias en este campo. Al mismo tiempo, es importante desarrollar relaciones a largo plazo con los clientes, puesto que los proyectos de Internet son proyectos vivos, que requieren mantenimiento y servicios de gestión y presentan continuas oportunidades de mejoras y futuros proyectos en clientes ya existentes.



personas



procesos



tecnologías

Paradigma de Calidad

La Gestión de la Calidad en el desarrollo del software se basa en una organización bien definida, con los medios, metodología y herramientas adecuadas para realizar una adecuada gestión de proyectos, complementada por una correcta planificación y seguimiento de la Calidad en cada proyecto. Es necesaria una dirección consciente y comprometida, que lidere eficazmente todo este sistema de trabajo, y un personal formado y convencido de que **la calidad se genera en el trabajo diario y no en la inspección o control del producto final.**

La implantación de un sistema de calidad es una cuestión de largo plazo, no de corto, y no es simplemente un conjunto de procedimientos; es una filosofía y una práctica de gestión empresarial y es vital para hacer eficaz el trabajo.

Sin el compromiso serio de la dirección, el proyecto carece de sentido. Por ello, la definición de un Plan Estratégico, formular la Política de Calidad y constituir la estructura de calidad, conformara no sólo un Sistema de Aseguramiento de la Calidad, Sino que dará forma a lo que se puede denominar como Paradigma de la calidad empresarial.

Para implementar este paradigma en una organización son necesarias las siguientes etapas y actividades:

Etapa 1

Elaborar el Plan estratégico: Etapa preliminar fundamental para el éxito del proyecto ya que se define la documentación estratégica, básica y de soporte, se formulan aspectos críticos, políticas y estructuras básicas, procesos esenciales y de valor añadido, se determina el alcance real e impacto del proyecto, se realiza la planificación de detalle y se lanza y comunica el proyecto.

La profundidad y el rigor de la realización de las actividades contempladas es vital para determinar el grado real de compromiso de la dirección y el nivel de involucración del personal –directivo y técnico- de la organización.

Deberá definirse de una forma clara y unívoca la dirección a seguir y especificar explícitamente la misión y los objetivos de la División, así como la organización y los medios para alcanzar dichos objetivos.

En primer lugar se debe establecer de una forma clara la posición de la Calidad, su relevancia para el cumplimiento de los objetivos estratégicos generales, las relaciones con otras divisiones, empresas y departamentos con los que se relacione profesionalmente, y su autonomía e independencia, y en segundo lugar, se deben definir todos los objetivos estratégicos y tácticos –en concreto los relacionados con la calidad de los productos y del servicio- a alcanzar en el sector al que se dirige y en el que opera, las prioridades de actuación y los criterios operativos, la cadena de valor, la tipología de clientes, las líneas de productos y servicios sobre las que se va a trabajar, la estrategia organizativa, los diferentes planes operativos –de I+D e Innovación, de Calidad, Comercial, de Diseño y Desarrollo, de Recursos

Humanos, de Inversiones, de Financiación, etc.- y establecer el calendario de implantación..

Etapa 2

Elaboración del manual de organización: Debe ser una consecuencia directa del Plan de Recursos Humanos del Plan Estratégico. El **Manual de Organización** debe regir todos los procesos administrativos, comerciales y de personal de la empresa.

Debe estar estructurado según procedimientos de organización y en él se deben establecer niveles que requieren unas aptitudes y conllevan unas funciones, e integra a todo el personal de la empresa. Este manual debe contemplar claramente el seguimiento y control de gestión, la estructura organizativa, la definición de puestos de trabajo y su valoración, el inventario de personal (incluyendo conocimientos y experiencias), el seguimiento de la actividad, los mecanismos y procedimientos de selección, los mecanismos de evaluación del trabajo realizado, la administración contable, la adquisición de bienes, la contratación de servicios, la administración de personal y la seguridad de equipos e instalaciones.

Etapa 3

Formular la Política de calidad: Definición, documentación y difusión de la **política** de calidad, incluyendo los **objetivos** y el **compromiso** en materia de calidad. La política de calidad debe ser adecuada a los objetivos de la división y a las expectativas y necesidades de los clientes, tanto internos, como externos a ella.

Esta política debe ser entendida, implantada y mantenida al día en todos los niveles de la Organización

La responsabilidad de la dirección en la política de calidad y en las actividades de verificación debe ser definida tanto respecto al Aseguramiento de la Calidad del software como a la Planificación y al Seguimiento y Control del proyecto.

Etapa 4

Definición de la estructura de calidad. **Será** responsabilidad de toda la organización el asegurar que los requisitos que se describan en el Sistema de Aseguramiento de la Calidad son seguidos por todo el personal; para ello la Dirección de la empresa velará por el cumplimiento del sistema e intervendrá cuando aparezcan problemas de calidad que no se han resuelto en los niveles inferiores de la organización.

La calidad del trabajo es responsabilidad de la persona que lo ejecuta. Para reforzar su aseguramiento, la empresa definirá actividades de prevención y control ligadas a una organización con capacidad para aceptar o rechazar los trabajos y corregir las desviaciones observadas. La organización se apoyará en la inequívoca definición en los procedimientos de responsabilidades asociadas a la ejecución de las diferentes tareas de prevención y control. La organización de las actividades del Sistema de Calidad se debe apoyar en la estructura organizativa general de la empresa compuesta por niveles a los cuales se les asignan funciones y requisitos determinados en el Manual de Organización de la empresa.

La Dirección es responsable de velar por la revisión del Sistema de Aseguramiento de la Calidad establecido, en los intervalos fijados, en base a la información interna sobre efectividad y eficacia de las acciones que se describan para el aseguramiento de la calidad.

Etapa 5

Definir los procesos. Un proceso, en general, es un conjunto de actividades encadenadas, necesarias para conseguir un propósito o para cumplir un objetivo. Siempre producen unos resultados a partir de unas entradas dadas. Por ejemplo, el proceso de “diseño de software” está compuesto por todas las actividades necesarias para desarrollar un diseño de software que se adecúe eficazmente a los requisitos especificados.

En esta actividad se deberán definir y describir los **procesos** que en la empresa son necesarios realizar para la adquisición, desarrollo, suministro, instalación, operación, actualización y soporte del software y de las aplicaciones y sistemas desarrollados, así como las **prácticas** genéricas que caracterizan el nivel de capacidad de los procesos.

El propósito es la construcción de una librería de procesos críticos básicos (incluyendo su definición, estándares, procedimientos y modelos). Esta librería servirá para evaluar la capacidad y eficacia de todos los procesos definidos que soportarán posteriormente un estado estable y repetible del proceso global de ingeniería y gestión del software.

Etapa 6

La sensibilización. La mejor manera de introducir la calidad como hábito de trabajo es la formación, tanto en el ámbito global de organización como en los equipos de trabajo.

En esta primera etapa (**sensibilización**), la formación tratará sobre aspectos fundamentales y generales de la gestión de la calidad, necesidad y beneficios de la implantación de un Sistema de Aseguramiento de la Calidad, conceptos y métodos básicos, modelos, herramientas y métodos de calidad, con un enfoque y una particularización a la mejora de procesos de software.

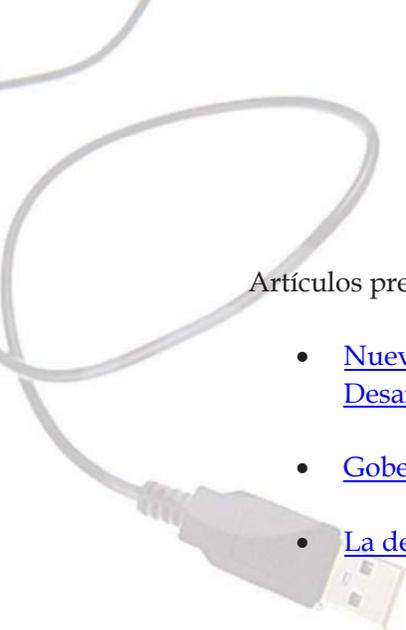
Se hará una introducción a mecanismos mentales y lógicas de gestión básicos como: relación causa-efecto, ciclo PDCA, métricas generales en procesos de software, herramientas estadísticas básicas, herramientas estadísticas avanzadas y algunas técnicas de fiabilidad.

La implantación de un sistema de calidad es una cuestión de largo plazo, no de corto, y el Sistema de Gestión de la Calidad tiene que considerar las tres dimensiones, control, mejora y aseguramiento y no puede improvisarse: tiene que existir una **planificación**, un **control** y unas **acciones de mejora**.

Antes de comenzar la ejecución de cualquier proyecto estratégico para la empresa es necesario difundir entre todo el personal el proyecto y sus objetivos, con el fin de conseguir el máximo compromiso y de que todo el personal de la empresa se sienta involucrado en el mismo.

Para ello es necesario comunicar que es un proyecto de todos y en el que todos deben participar, ya que no se trata de tener una gran cantidad de manuales y procedimientos sino de establecer e institucionalizar mecanismos que faciliten el trabajo diario y favorezcan la labor creativa de todas las personas de la división.

¡Es un proyecto de todos y todos deben sentirse y estar involucrados!



Artículos presentados en esta selección:

- [Nuevos retos, viejos problemas El camino hacia una cultura de 'Excelencia en el Desarrollo de Software'](#)
- [Gobernador de TI, Nuevo garante de la calidad de las TIC dentro de la Organización.](#)
- [La descentralización de los proyectos de Software.](#)



Nuevos retos, viejos problemas El camino hacia una cultura de 'Excelencia en el Desarrollo de Software'

Artículo



Hoy en día términos como "nearshore" y "offshore" son totalmente comunes. Incluso sus traducciones al español como "factoría de software", "centro de servicio" o "centro de alta tecnología" son usualmente utilizadas. Prácticamente, casi todo el mundo tiene su factoría propia, aunque sólo se trate de 15 empleados desplazados en otra provincia o en otro edificio de la misma ciudad.

Parece ser que este modelo de desarrollar software va a ser el utilizado en los próximos años por la mayoría de las grandes

empresas, pero ¿realmente estamos interesados en este modelo como una acción más para tratar de convertir en industria el sector del software, o lo único que nos interesa es que nos desarrollen software a bajo coste?

Indudablemente la ventaja económica que ofrece este modelo es muy atractiva, pero ¿estamos preparados para industrializar nuestra manera de desarrollar software? ¿Somos conscientes de que aún no hemos madurado como industria? Estos dilemas se deben plantear si realmente de lo que se trata es de cambiar la manera de trabajar y no de trasladar los problemas que actual-

68 | calidad - julio-agosto 2007

Autor: Domingo Gaitero Gordillo Managing Consultant de Atos Origin SAE. Vicepresidente del Comité de Software de la AEC

Publicado en la Revista CALIDAD Nº 5, JULIO-AGOSTO 2007

Hoy en día términos como "nearshore" y "offshore" son totalmente comunes. Incluso sus traducciones al español como "factoría de software", "centro de servicio" o "centro de alta tecnología" son usualmente utilizadas. Prácticamente, casi todo el mundo tiene su factoría propia, aunque sólo se trate de 15 empleados desplazados en otra provincia o en otro edificio de la misma ciudad.

Parece ser que este modelo de desarrollar

software va a ser el utilizado en los próximos años por la mayoría de las grandes empresas, pero ¿realmente estamos interesados en este modelo como una acción más para tratar de convertir en industria el sector del software, o lo único que nos interesa es que nos desarrollen software a bajo coste?

Indudablemente la ventaja económica que ofrece este modelo es muy atractiva, pero ¿estamos preparados para industrializar nuestra manera de desarrollar software? ¿Somos conscientes de que aún no hemos madurado como industria? Estos dilemas se deben plantear si realmente de lo que se trata es de cambiar la manera de trabajar y no de trasladar los problemas que actualmente tenemos unos cientos de kilómetros.

La distancia es un tema que todavía no se ha superado. Tenemos una cultura de comunicación hablada muy grande, lo cual dificulta notablemente esta nueva manera de producir software. Llevamos demasiado tiempo haciendo tomas de requisitos, análisis funcionales y, sobre todo, extraordinarios diseños de modo "verbal", pero con este nuevo modelo la comunicación debe ser escrita.

A este problema de comunicación le podemos añadir alguno más como es la "extraña vinculación" que existe entre el profesional informático con el código y la acción de programar; parece que no se hiciera otra cosa, cuando no es así.

Asimismo, la necesidad de conocer a las personas que van a programar es otro obstáculo que suele aparecer por parte de algunos clientes. Sin embargo, el más peligroso de todos es el metodológico.

Por diferentes motivos que no vienen a cuenta (el "time to market", el día a día, los

presupuestos, la rotación de personal...) se ha llegado a que casi todos los desarrollos se lleven de una manera más cercana a la artesanal que a la industrial; basándose en los profesionales, estupendos por cierto, que son sobre los que recae todo el peso del proyecto en la mayoría de los casos, y de los que depende generalmente su éxito o fracaso.

Por ello es necesario que las organizaciones antes de lanzarse a nuevos modelos revisen una asignatura pendiente que, parece ser, tenemos suspendida eternamente los profesionales y, por lo tanto, las organizaciones, del sector: la ingeniería del software.

Toda organización debe tener su propia cultura, un conjunto de creencias, objetivos y valores comunes que caracterizan sus prioridades y acciones. La cultura de un grupo dedicado al desarrollo de software repercute decisivamente sobre la calidad de sus productos, la productividad de sus desarrolladores y el trabajo en equipo de sus miembros.

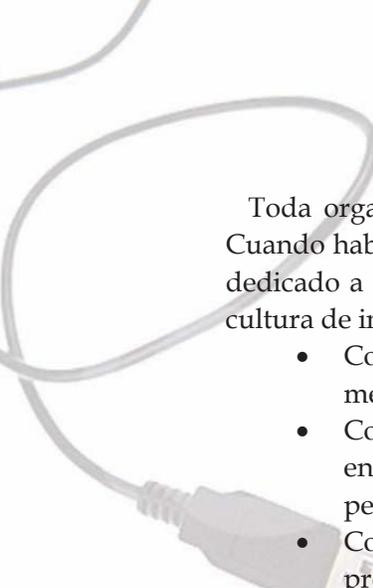
Un desarrollador de software comprometido con la calidad tiene una "cultura en el desarrollo software" según la cual determinadas prácticas directivas y técnicas propician más que otras la creación de productos de calidad en un entorno positivo.

En general, estas técnicas representan "un mínimo" de las prácticas tradicionales completas de la ingeniería de software. Una cultura de ingeniería de software supone la fusión de las actitudes orientadas a la calidad, las interacciones humanas y los procesos técnicos. Se trata de una cultura basada en ciertos principios –a menudo subyacentes– que guían la conducta de los integrantes de la organización.

El diccionario define "cultura" como "la totalidad de los patrones de conducta, artes, creencias, instituciones y demás productos resultantes del trabajo y del pensamiento humanos característicos de una comunidad o población y que se transmiten socialmente". La cultura incluye un conjunto de valores, objetivos y principios compartidos que dirigen las conductas, actividades, prioridades y decisiones de un grupo de personas que trabaja por un objetivo común.

La capacidad de introducir cualquier cambio específico en las técnicas o herramientas utilizadas por una organización depende de las actitudes culturales predominantes. Un personal de desarrollo comprometido con la ingeniería de software es más receptivo a probar nuevos enfoques que puedan contribuir a cumplir este objetivo. Por el contrario, con la actitud de "escribamos el código porque no tenemos tiempo de pensar sobre el problema antes de intentar resolverlo", el director lo tiene difícil para promover mejoras en el proceso o cambios de conducta.





Toda organización tiene su propia cultura, pero algunas son más estimulantes que otras. Cuando hablamos de “cultura de excelencia en el desarrollo software”, nos referimos al grupo dedicado a la calidad, no a quienes escriben el código. A menudo aludimos a ella como una cultura de ingeniería de software “sana”, puesto que incluye tres componentes esenciales:

- Compromiso personal de todos los desarrolladores para crear productos de calidad mediante la aplicación sistemática de técnicas de ingeniería de software eficaces.
- Compromiso corporativo de los directores en todos los niveles para propiciar un entorno donde la calidad del software sea un factor decisivo para el éxito y se permita a todos los desarrolladores cumplir este objetivo.
- Compromiso de todos los miembros del equipo para mejorar constantemente los procesos que utilizan y así mejorar continuamente los productos que crean.

De igual modo, la cultura es un punto de partida para decisiones y acciones, cuya aplicación constante refuerza más aún los valores culturales. La cultura también ayuda a los directivos de la organización a determinar sus prioridades. La transmisión de unas prioridades consistentes por parte de los directores transmite el mensaje de que todos los niveles de la organización comparten compromiso con la calidad del software. Estos ciclos de retroalimentación sirven de refuerzo y son necesarios para mantener la evolución de la cultura del software hacia un rendimiento cada vez mejor.

Quienes trabajan en una cultura orientada hacia la ingeniería del software saben que el desarrollo de aplicaciones es mucho más que escribir programas. Las tareas de gestionar y planificar proyectos, gestionar requisitos, analizar y diseñar programas, validación, verificación y documentación superan con creces el esfuerzo empleado en el paso de la programación.

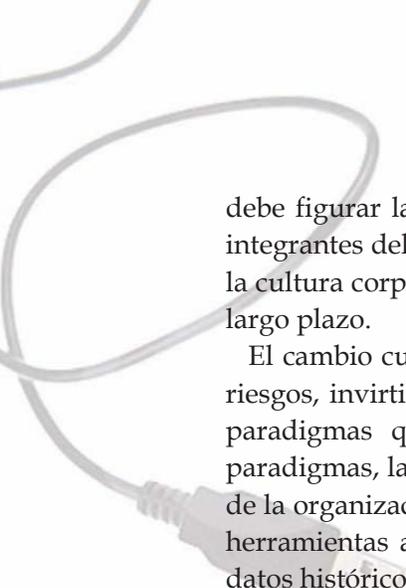
Estas otras actividades asientan las bases para un código íntegro que ofrezca al cliente lo que éste necesita. Una buena “cultura de excelencia del desarrollo software” se centra en desarrollar aplicaciones útiles mediante el uso de diversas herramientas y técnicas que cubran toda una serie de problemas sobre aplicaciones, no simplemente la escritura de programas.

De manera consciente o inconsciente, en una organización de software todos seguimos algún proceso para desarrollar software, alguna secuencia de acciones que nos llevan desde el concepto inicial hasta el código ejecutable.

Este proceso es el conjunto de procedimientos, métodos y herramientas que utilizamos para crear software. Las especificaciones de requisitos de software suelen contener tantos errores que no siempre podemos saber si estamos construyendo el producto correcto. Necesitamos procesos que generen productos de calidad en cada fase del desarrollo, no sólo en la codificación. Los desarrolladores de software suelen aplicar técnicas de calidad para inspeccionar los productos o someterlos a pruebas prácticas en busca de errores. Debemos, además, aplicar las técnicas de calidad para mejorar y controlar los demás pasos del proceso y así prevenir defectos.

La cultura de una organización es una fusión de su gente, sus procesos y sus tecnologías. Si se desatiende alguno de estos aspectos disminuirá el potencial impacto de los cambios que intente introducir. Los procesos que se aplican de manera esporádica, ineficaz o inapropiada pueden hacer más mal que bien. Las tecnologías desaprovechadas o poco utilizadas suponen un malgasto de dinero, y las personas cuyo desarrollo profesional se descuida seguramente buscarán oportunidades de empleo en otra parte.

Puesto que los procesos los llevan a cabo personas, los miembros de nuestro equipo siempre serán su activo máspreciado. Y por consiguiente, entre los esfuerzos de mejora de su grupo



debe figurar la consolidación de capacidades, conocimiento y entorno laboral para todos los integrantes del equipo. Para convertirse en un agente de cambio eficaz, debe trabajar dentro de la cultura corporativa existente, al tiempo que luchar por modificarla para generar beneficios a largo plazo.

El cambio cultural se puede promover recompensando a las personas por asumir pequeños riesgos, invirtiendo en la aplicación de nuevos procesos y tecnologías y superando los viejos paradigmas que limitan el potencial de rendimiento de la organización. Entre dichos paradigmas, las tecnologías obsoletas instauradas que han dejado de satisfacer las necesidades de la organización, la ejecución manual de las pruebas y de las tareas de diseño cuando existen herramientas automatizadas, así como la planificación informal de proyectos sin basarse en datos históricos de otros anteriores.

Para instaurar cambios en los procesos, hay que tener en cuenta cuatro principios:

- La mejora de los procesos debe ser evolutiva, continua y cíclica.
- El cambio debe ser tanto de las organizaciones como de los individuos, pero las personas solamente cambian cuando tienen un motivo para ello.
- Los cambios en los procesos han de orientarse hacia los objetivos, basarse en datos y evitar el dogmatismo.
- Las actividades de mejora deben considerarse pequeños proyectos.

La mejora del proceso de software para convertirse en “excelencia en el desarrollo de software” puede dar mucho trabajo, pero también resultarle muy gratificante mientras ayuda a dirigir la cultura de su equipo hacia un mayor rendimiento y un mejor lugar de trabajo. Una vez que esta cultura esté institucionalizada en la organización, bienvenidos sean los nuevos modelos porque ya habremos dado un primer paso hacia la verdadera industrialización del software.

Para terminar sería bueno reflexionar sobre una frase que escribió Stephen Covey, y que puede ayudarnos notablemente:

“Si seguimos haciendo lo que estamos haciendo, seguiremos consiguiendo lo que estamos consiguiendo.”

Gobernador de TI, Nuevo garante de la calidad de las TIC dentro de la Organización



Nuevo garante de la calidad de las

Los Consejos de Administración y las Direcciones Generales han comprendido, desde hace tiempo, la necesidad de establecer marcos de buen gobierno corporativo. Esto se hace particularmente evidente si se tienen en cuenta las crecientes obligaciones en materia de conformidad normativa.

Más que los Sistemas de Información —las Tecnologías de la Información (TI) y las Comunicaciones que los sustentan— han adquirido una gran relevancia en el logro de los objetivos corporativos y en la entrega de beneficios. Esto ha llevado a un importante número de organizaciones a darse cuenta de que la gobernanza no se extiende también al ámbito tecnológico hasta tal punto que, hoy en día, el buen gobierno de las TI es considerado una parte integrante de la gobernanza corporativa, como medio de apoyar y reforzar las estrategias y objetivos de la organización.

Este escenario —con una creciente demanda por parte del negocio, de la aportación que ofrece la tecnología y con una superior concienciación sobre la importancia de contar con buenas prácticas y modelos— abre un claro camino hacia el desarrollo y despliegue de la Gobernanza de TI.

Como parte de este “despliegue”, y dado que la dirección (gobierno) de las TI ya no ha de verse como una preocupación exclusiva de la Dirección de Informática, sino de la alta dirección en su conjunto, está surgiendo una nueva figura a nivel directivo: el Gobernador corporativo de TI (Chief [IT] Governance Officer, o CGO, como recogería la terminología anglosajona).

Como ventaja añadida, el establecimiento de un puesto tal dentro de la organización supondrá una demostración palpable del compromiso de aquella con la excelencia, en las buenas

prácticas de gobernanza de TI y, por ende, con la calidad de los servicios prestados desde el área de tecnología.

Con la mira puesta en el negocio

En el nuevo contexto, la misión del gobernador de TI será proporcionar el debido apoyo al Consejo de Administración y a la Dirección General, maximizando la calidad de las TI y su contribución al éxito de la organización y, al mismo tiempo, gestionando y mitigando los riesgos derivados de la propia tecnología.

El gobernador de TI deberá, igualmente, armonizar el valor de los marcos de referencia y metodologías de dirección de TI, o intentar y lograr el establecimiento de buenas prácticas para el gobierno, la gestión y el control de las TI, a lo largo y ancho de la organización.

El perfil

Entre las responsabilidades —o, dicho de otro modo, habilidades— de los profesionales que desarrollen su actividad dentro de la gobernanza de las TI en sus organizaciones, cabe destacar las siguientes:

• Deberán ser capaces de asegurar que las TI actúan como catalizador para el logro de los objetivos del negocio, mediante la integración de los planes estratégicos de TI con los planes estratégicos de la organización y, mediante la sincronización de los servicios prestados desde TI, con los operativos de la compañía, para optimizar los procesos de negocio. Todo ello, a través del desarrollo de una estrategia tecnológica de la empresa.

Autor: Miguel García Menéndez CISM, CISA
Vocal del Comité de Software de la AEC Asesor
sobre Gobernanza de TI en Atos Consulting

Publicado en la Revista CALIDAD N° 1,
FEBRERO 2008.

Los Consejos de Administración y las Direcciones Generales han comprendido, desde hace tiempo, la necesidad de establecer marcos de buen gobierno corporativo. Esto se hace particularmente evidente si se tienen en cuenta las crecientes obligaciones en materia de conformidad normativa.

Más aún, los Sistemas de Información —las Tecnologías de la Información (TI) y las Comunicaciones que los sustentan— han

adquirido una gran relevancia en el logro de los objetivos corporativos y en la entrega de beneficios. Ello ha llevado a un importante número de organizaciones a darse cuenta de que la gobernanza ha de extenderse también al ámbito tecnológico; hasta tal punto que, hoy en día, el buen gobierno de las TI es considerado una parte integrante de la gobernanza corporativa, como medio de apoyar y reforzar las estrategias y objetivos de la organización.

Este escenario —con una creciente demanda por parte del negocio, de la aportación que ofrece la tecnología y con una superior concienciación sobre la importancia de contar con buenas prácticas y modelos— abre un claro camino hacia el desarrollo y despliegue de la *Gobernanza de TI*.

Como parte de este “despliegue”, y dado que la dirección (gobierno) de las TI ya no ha de verse como una preocupación exclusiva de la Dirección de Informática, sino de la alta dirección en su conjunto, está surgiendo una nueva figura a nivel directivo: el *Gobernador corporativo de TI (Chief [IT] Governance Officer, o CGO, como recogería la terminología anglosajona)*.

Como ventaja añadida, el establecimiento de un puesto tal dentro de la organización supondrá una demostración palpable del compromiso de aquella con la excelencia, en las buenas prácticas de gobernanza de TI y, por ende, con la calidad de los servicios prestados desde el área de tecnología.

Con la mira puesta en el negocio

En el nuevo contexto, la misión del gobernador de TI será proporcionar el debido apoyo al Consejo de Administración y a la Dirección General, maximizando la calidad de las TI y su contribución al éxito de la organización y, al mismo tiempo, gestionando y mitigando los riesgos derivados de la propia tecnología.

El gobernador de TI deberá, igualmente, armonizar el valor de los marcos de referencia y metodologías de dirección de TI, e impulsar y apoyar el establecimiento de buenas prácticas para el gobierno, la gestión y el control de las TI, a lo largo y ancho de la organización.

El perfil

Entre las responsabilidades – o, dicho de otro modo, habilidades – de los profesionales que desarrollen su actividad entorno a la gobernanza de las TI en sus organizaciones, cabe destacar las siguientes:

- Deberán ser capaces de asegurar que las TI actúan como catalizador para el logro de los objetivos del negocio, mediante la integración de los planes estratégicos de TI con los planes estratégicos de la organización y, mediante la sincronización de los servicios prestados desde TI, con las operaciones de la compañía, para optimizar los procesos de negocio. Todo ello, a través del *desarrollo de una estrategia tecnológica de la empresa*.
- Deberán, asimismo, encargarse de garantizar que, tanto las TI, como las áreas de negocio, cumplen con sus responsabilidades sobre la gestión del valor: esto es, que las nuevas inversiones en actividades apoyadas en tecnología producen el beneficio esperado y aportan un valor al negocio, medible tanto individual, como colectivamente; que las capacidades (soluciones y servicios) son entregados en tiempo y coste; y que los servicios y otros activos de TI contribuyen de manera continua al valor del negocio. Todo ello, mediante el *desarrollo de un proceso de gobierno del valor*.
- Adicionalmente, los gobernadores de TI deberán asegurar la existencia y puesta en marcha de marcos apropiados, alineados con las normas y modelos de referencia, para identificar, evaluar, mitigar, gestionar, comunicar y supervisar los riesgos del negocio relacionados con las TI, como una actividad más del buen gobierno de la empresa. Todo ello, mediante el *desarrollo, mejora y mantenimiento de un proceso continuo y analítico de gestión de los riesgos empresariales*.
- Estos nuevos directivos deberán ocuparse de que el área de TI disponga de recursos suficientes, competentes y capaces de ejecutar los objetivos estratégicos presentes y futuros, así como de responder a las demandas del negocio, a través de una optimización de la inversión, del uso y de la ubicación de los activos tecnológicos (aplicaciones, información, infraestructuras y personal). Todo ello, mediante la *puesta en marcha de un proceso continuo de planificación, gestión y evaluación de recursos*.
- Finalmente, deberán asegurar que se establecen metas e indicadores para las TI, de apoyo al negocio y en colaboración con las partes interesadas; y que se fijen, supervisen y evalúen ciertos objetivos medibles. Todo ello, mediante *procesos continuos de gestión y evaluación del rendimiento*.

Se trata, en definitiva, de habilidades y actividades directamente relacionadas con la

definición, el establecimiento y/o el mantenimiento de un marco de referencia para la gobernanza de las TI —materializada en liderazgo, estructuras organizativas y procesos— para: asegurar el alineamiento con el buen gobierno de la entidad; controlar el entorno de TI y la información del negocio mediante la puesta en marcha de buenas prácticas y garantizar la conformidad con los requisitos externos.



Una reflexión final

Pero, acaso, el perfil descrito ¿se corresponde solamente al esperado de un directivo encargado del buen gobierno de las TI, dentro de su organización?; o, de hecho, ¿no se tratará también, de un catálogo de competencias con las que debería contar toda la comunidad de profesionales cuya actividad se desarrolle en los ámbitos de la calidad de las TI y de la mejora de los procesos subyacentes al ciclo de vida de las TIC?



personas



procesos



tecnologías

La descentralización de los proyectos de Software



Autor Luisa Morales Gómez-Tejedor

Publicado en la Revista Calidad - Diciembre 2004 – Enero 2005

Factores como la competencia, el time to market y los niveles cada vez más exigentes de calidad, están impulsando a las empresas a buscar soluciones imaginativas encaminadas a maximizar los niveles de calidad de sus productos y servicios, reduciendo a la vez los costes de los mismos. Un factor clave en este terreno es el cambio que se está produciendo en el enfoque de los proyectos tradicionales, abordados en una única localización y con un mismo equipo de trabajo, hacia un enfoque descentralizado con un equipo multidisciplinar donde prima la especialización. Con estos ingredientes, la clave de la descentralización está en encontrar los grupos de mayor especialización en las localizaciones de menor coste.

En el proceso de desarrollo de software hay tareas que necesariamente deben hacerse en el centro de la empresa, donde están ubicados los usuarios que definirán el sistema de información, la máquina sobre la que correrá una vez puesto en funcionamiento, o los técnicos que recibirán el sistema para su implantación y mantenimiento.

Hay, sin embargo, otras tareas que no tienen una dependencia tan alta de los elementos inherentes al centro de la empresa y que pueden abordarse desde otros centros, que, si bien están más alejados en distancia, también están más especializados en el propio diseño y construcción de los sistemas. Los centros de desarrollo a distancia cuentan con entornos de desarrollo estables (hardware, software, herramientas, soporte)

así como infraestructura tecnológica y espacio de trabajo operativos desde el primer día del proyecto. A esto se unen dos ingredientes fundamentales, un equipo estable de profesionales formados en la metodología del centro de desarrollo y un modelo de funcionamiento basado en procesos consistentes, medibles y, sobre todo, repetibles. Como resultado, estos centros consiguen un incremento de la productividad, manteniendo un nivel alto de calidad. Cuando se utiliza un centro de desarrollo a distancia como alternativa al desplazamiento de profesionales al centro de la empresa, a las ventajas señaladas se une la reducción del coste por dos factores: los desplazamientos (viajes, hoteles, dietas) y el propio espacio físico del centro que se libera.

Un segundo análisis de las tareas del proceso de desarrollo nos conduce a ver que existen distintos grados de dependencia del centro de la empresa, de los usuarios fundamentalmente. En el análisis, por ejemplo, se necesita mantener una comunicación fluida con los usuarios para garantizar la correcta interpretación de los requerimientos. El resultado del análisis, sin embargo, es un conjunto de especificaciones técnicas fácilmente interpretables por especialistas en construcción de software. Esto nos lleva nuevamente a una separación de tareas: el análisis y puesta en marcha puede hacerse por un grupo de analistas cuya especialidad es

garantizar la correcta interpretación de los requerimientos. El resultado del análisis, sin embargo, es un conjunto de especificaciones técnicas fácilmente interpretables por especialistas en construcción de software. Esto nos lleva nuevamente a una separación de tareas: el análisis y prueba del sistema puede hacerse por un grupo de analistas cuya especialidad es la funcionalidad, mientras que la codificación y prueba unitaria pueden llevarse a cabo por equipos de programación con una especialidad eminentemente técnica. Por lo que, en determinados casos, se puede llegar a hablar de dos tipos de centros de desarrollo a distancia, con distintos grados de cercanía. El primero, más enfocado a la funcionalidad del sistema, tiene sentido que tenga un mayor grado de proximidad, mismo país o misma franja horaria, mientras que el segundo, que recibirá especificaciones y devolverá lotes de programas, puede estar ubicado allí donde los costes sean realmente bajos. En esta línea, está siendo creciente la demanda de centros de desarrollo ubicados en la India, Filipinas o China.

Localización de equipos especializados según las etapas del proceso de desarrollo

DEFINICIÓN DE REQUISITOS	ANÁLISIS	CODIFICACIÓN Y PRUEBA UNITARIA	PRUEBA DEL SISTEMA	ACEPTACIÓN E IMPLANTACIÓN
		Centro de desarrollo nivel 2		
	Centro de desarrollo nivel 1			
Centro de la empresa				

El mejor enfoque de descentralización no existe, cada caso requiere un análisis y una decisión que puede consistir unas veces en llevar a un centro de desarrollo a distancia exclusivamente la codificación y prueba unitaria, otras por el contrario en llevar el mayor número de tareas a un centro de desarrollo, o también en distribuir el trabajo en varios centros, según criterios de especialización y coste. En el cuadro se muestra una distribución del trabajo en tres niveles, basándose en las etapas del proceso de desarrollo, también podría hacerse la distribución basándose en un enfoque funcional, por ejemplo, llevando a otro centro la migración de los datos relacionados con el sistema que se está desarrollando.

Una forma distinta de enfocar un proyecto requiere también una forma distinta de gestionarlo. Si bien las metodologías de desarrollo son bastante estrictas a la hora de definir los puntos de entrada y salida de cada tarea, en un enfoque distribuido en varios centros, los puntos de corte son extremadamente importantes, deben estar muy bien definidos y ser conocidos por todos. Los aspectos de mayor relevancia para la correcta gestión de este tipo de proyectos son:

- **Establecer los interlocutores válidos** La comunicación se establece en varios niveles, por ejemplo, recepción y entrega de trabajos, seguimiento de la planificación, resolución de

dudas funcionales, aspectos técnicos, etc. Cada nivel de comunicación requiere la identificación de un interlocutor válido en cada centro de trabajo. Este interlocutor es conocido por todos los centros.

- **Definir el flujo de comunicación** Cada integrante de los distintos equipos de los centros de desarrollo conoce el flujo que ha de seguir la comunicación. Los flujos más sensibles son los relacionados con los cambios, especialmente los cambios en las especificaciones y en el modelo de datos.
- **Definir los mecanismos de seguimiento y control** Cuando se envía trabajo a otros centros, aumenta la importancia de la información transparente y a tiempo. Es importante que se acuerde al principio del proyecto los diferentes mecanismos tanto de información como de seguimiento y control.
- **Hacer un plan de envíos y entregas** La forma más efectiva de funcionar con un centro de desarrollo a distancia es a través de la definición de paquetes de trabajo (grupos de programas, por ejemplo), estableciendo las fechas de envío al centro de los paquetes de diseños y las fechas de entrega de los programas por parte del centro. El plan permite estimar la carga trabajo y responder adecuadamente frente a puntas de trabajo elevadas.
- **Establecer criterios de entrada/salida** Una vez definidos los puntos de corte, se establece cómo se recibe y entrega el trabajo en cuanto a forma, contenido, cantidad y calidad. Por ejemplo, las especificaciones funcionales recibidas en el centro son revisadas y aceptadas por éste antes de iniciar el proceso de desarrollo, esto permite disminuir los errores derivados de malas especificaciones, cuyo coste de reparación es elevado ya que se detectan después de la entrega de los trabajos.
- **Definir la gestión de la configuración** La figura del gestor de la configuración crece en importancia en este tipo de proyectos en los que no sólo hay que definir distintos entornos teniendo en cuenta las fases del proyecto (varios entornos de desarrollo, un entorno de pruebas de ensamblaje, de prueba integrada, etc.), sino que también hay que considerar los distintos centros involucrados, así como las librerías de entrada y salida para el envío y recepción de elementos. Junto con la definición de entornos se establecen también las normas del movimiento del software entre los distintos entornos.



La descentralización de los proyectos de software a través de la especialización de equipos de trabajo integrados de una manera lógica, que no física, con una forma estándar de interactuar entre los mismos, está siendo la respuesta a las exigencias de un mercado cada vez más competitivo, en el que prima maximizar la eficiencia manteniendo alto tanto el nivel de calidad como el tiempo de respuesta o puesta en el mercado del nuevo producto.



personas



procesos



tecnologías

Medidas

Este es el grupo más numeroso y es que ya se sabe que lo que no se mide no mejora, por eso desde este comité siempre se ha dado una importancia relevante al tema de la medición en todos sus posibles ámbitos: “modelos para estimaciones de proyectos software”, “evaluación de proveedores mediante la gestión de un repositorio de proyectos”, “las técnicas de los puntos de función” son algunos de los seis artículos que mostramos.

Para mejorar continuamente nuestros productos o servicios TIC es necesario poder evaluarlos; teniendo en cuenta que no se puede mejorar lo que no se puede medir y que además, para medir cualquier factor es absolutamente esencial contar con datos correctos, válidos y fiables.

Por tanto, el seguimiento y la medición son necesarios para conocer el nivel de cada factor en cada momento y poder compararlo con situaciones precedentes. Deben definirse indicadores que permitan medir la calidad tanto de los productos como de los procesos y, tanto internamente (cliente interno) como desde el punto de vista del cliente (cliente externo), para asegurar el desempeño de la organización, así como la satisfacción de las partes interesadas.

La medición en sí misma no mejora el proceso pero, por un lado, la visibilidad de datos proporcionada por la medida da la capacidad para la toma de decisiones cuantificadas, y por otro lado, permite comprobar el grado de mejora conseguido. Esto será consecuencia del análisis de los valores o datos obtenidos de los indicadores.

Finalmente, el resultado del análisis deberá desencadenar las decisiones necesarias y oportunas para conseguir niveles de mejora en aquellos aspectos para los que se haya detectado esta necesidad. Así mismo, habrá que evaluar la efectividad de las medidas. Estas decisiones pueden provocar acciones correctivas y revisión de los procesos, así como acciones preventivas para evitar futuros problemas similares.

El modelo de medición debe ser coherente con la estrategia y visión de la empresa, por lo que debe extenderse a todas las actividades de la organización, ya que el objetivo final de este modelo será la mejora continua en todos los ámbitos de la misma.

La **organización** existe por las **personas** que desempeñan su labor, según unos determinados **procesos**, generando unos **productos** que satisfagan las necesidades de los **clientes**.

Por tanto, parece necesario seguir y medir:

- las competencias personales: actitudes, aptitudes y capacidades
- las competencias tecnológicas (visión, recursos y capacidades tecnológicas) y las competencias organizativas (aprendizaje organizativo y aptitudes y habilidades directivas) que deben reflejarse en los procesos de la organización (la medición deberá comprobar que los procesos efectivamente se ajustan a la visión tecnológica y organizativa de la organización)
- los productos: la calidad, la imagen, su aceptación en el mercado, etc.
- la ventaja competitiva: la satisfacción de los clientes, los aspectos financieros, en

definitiva la calidad, el precio y el tiempo.

Esto implica el seguimiento y medición de activos tangibles e intangibles.

La conformidad del producto y/o servicio con los requisitos se sigue y controla mediante las pruebas, inspecciones, revisiones y verificaciones y validaciones a lo largo de su ciclo de desarrollo.

Para la medida de los procesos el primer paso es identificar los procesos fundamentales y los de soporte. Los procesos fundamentales son los que atañen a diferentes áreas del servicio y/o desarrollo del producto y tienen impacto en el cliente creando valor para éste. Los procesos de soporte dan apoyo a los procesos fundamentales y ayudan a la hora de realizarlos. Todos los procesos seguirán las directrices fijadas por la Dirección y reflejadas en las Políticas de la Organización.

Identificados los procesos, y teniendo en cuenta que normalmente los procesos no se pueden medir de forma general, sino que hay que medir diferentes aspectos de los mismos, se deben definir criterios e indicadores para cada proceso. Una vez definidos todos los indicadores para cada proceso, se realizará un plan de recogida de datos, donde se explicitarán las fechas en que deben ser recogidos los datos de cada indicador así como la persona encargada de esa recogida.

La satisfacción del cliente se conseguirá, entre otros factores, proporcionando un buen producto y/o servicio. Sin embargo la organización debe intentar proporcionar un producto y/o servicio que satisfaga e incluso supere las expectativas del cliente. Será muy importante conocer la impresión del cliente en cuanto a los productos y servicios proporcionados, para poder mejorarlos si ha lugar.

Sin embargo, la medición tiene sentido dentro de la globalidad del proceso de mejora. Es necesario realizar un análisis de los datos obtenidos, pues estos por si solos y de manera independiente, no pueden explicar o predecir todo.

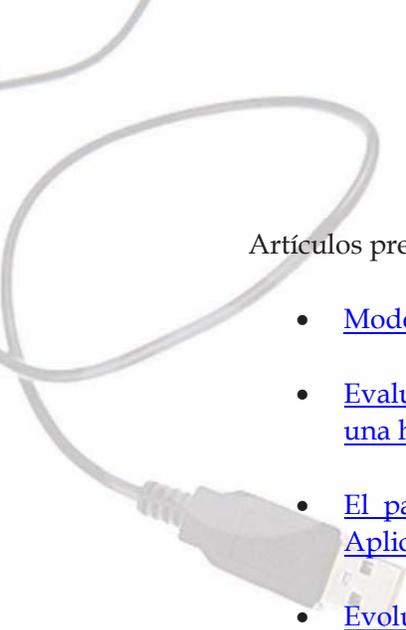
La organización recogerá datos procedentes principalmente del cálculo de los indicadores, aunque también de resultados de auditorías, encuestas de cliente, etc. Estos datos deben ser analizados para poder:

- Caracterizar los procesos y productos, de modo que se obtenga una línea base para futuras estimaciones
- Evaluar la consecución de los objetivos planteados anteriormente
- Predecir los objetivos alcanzables de calidad
- Identificar los aspectos susceptibles de mejora
- Validar la efectividad de las medidas tomadas

Datos determinantes para poder valorar la conformidad del producto, se obtendrán, principalmente, del análisis de las no conformidades; se deberá analizar utilizando la herramienta estadística más adecuada, la ocurrencia de no conformidades por tipo, severidad, frecuencia, fase en la que se originan, costes, costes de retrabajo, tiempos, etc.

Información sobre la satisfacción del cliente se obtendrá de los datos obtenidos de las encuestas. Información sobre los procesos se obtendrá de la monitorización de los indicadores definidos para cada proceso. Y datos sobre la implantación del sistema de aseguramiento de la calidad se obtendrá de las auditorías internas.

Pero todos estos datos deberán ser analizados de la manera más objetiva, repetible y documentada que sea posible. Para ello, se utilizarán herramientas estadísticas como pueden ser Hoja de recogida de datos, el Histograma, el Diagrama de Pareto, la Estratificación, el Diagrama causa-efecto, el Diagrama de correlación o las Hojas de control



Artículos presentados en esta sección:

- [Modelos para estimaciones del tamaño del software en proyectos.](#)
- [Evaluación de proveedores mediante la gestión de un repositorio de proyectos con una herramienta de estimación](#)
- [El papel de la técnica de Puntos Función en el 'Outsourcing' del Desarrollo de Aplicaciones](#)
- [Evolución de los planes de Métricas para el control de procesos de desarrollo de software desarrollo de software](#)
- [Indicadores de Calidad del Software: el valor de los números.](#)
- [Medición Práctica de Software y Sistemas \(PSM\) Información objetiva para la toma de decisiones.](#)



personas



procesos



tecnologías

Modelos para estimaciones del tamaño del software en proyectos.



Autor: Consuelo Lázaro Ruiz

Publicado en la Revista Calidad – Marzo 2004

Este artículo está basado en una de las ponencias realizadas el día 19 de noviembre de 2003 durante las Jornadas “El papel de los indicadores en la gestión del Software”, organizada por el Comité de Calidad del Software de la AEC. Los trabajos presentados en el transcurso de este encuentro trataron los temas de “Control del Desarrollo, Operación, Mantenimiento y Adquisición/Suministro del Software”. La ponencia utilizada como base de este artículo se enmarca dentro de la fase de Planificación, presentando algunos indicadores utilizados durante la fase de estudio y viabilidad de los proyectos.

Introducción

Uno de los principales problemas a los que se enfrenta cualquier empresa es el análisis de viabilidad de los posibles proyectos en su cartera. La decisión de acometer un proyecto determinado puede estar basada en muchos factores, pero es imprescindible conocer su “coste” y como no, el “beneficio” que obtendremos tras su realización. Dicho beneficio tangible, vendrá determinado por la facturación prevista, pero el análisis del coste es siempre menos inmediato.

Parte fundamental en la cuantificación del coste de un producto, es la valoración del “esfuerzo” necesario para su producción. Para conocer el esfuerzo se necesita “medir” lo que debe realizarse.

Las empresas que como SAES (S. A. de Electrónica Submarina) tienen como una de sus principales actividades el desarrollo de software, añaden a estos problemas en la valoración del esfuerzo, la dificultad de la realización de medidas sobre algo tan intangible como el software, teniendo que realizar “estimaciones” del tamaño y complejidad del software, en las etapas previas a la contratación. En estas etapas, el software aún no se encuentra totalmente definido, y únicamente se conocen los “requisitos funcionales del usuario”. Esto es, se sabe qué se le exige al software desde el punto de vista del usuario final, pero nada sobre su posible tamaño final.

Desde los años 60 hasta la actualidad, la industria del software se ha preocupado de

establecer distintos “modelos de estimación del tamaño del software” que faciliten la resolución de la ecuación viabilidad coste / beneficio, despejando la incógnita del tamaño.

FPA, Function Point Analysis. (Análisis de Puntos de Función)

El modelo de Análisis de Puntos de Función es quizás el más extendido en el mundo en lo que se refiere a realización de medidas de “tamaño funcional del software”.

La frase “Medida del Tamaño Funcional” se define en la “ISO/IEC 14143-1:1998 Functional Size Measurement - Definition of Concepts” como el proceso de medir el tamaño del software, derivado de cuantificar los Requisitos Funcionales de Usuario.

Desde 1979 en que Alan Albrecht de IBM introdujo el concepto básico de tamaño funcional, se ha avanzado mucho en este camino. Hoy en día se considera el tamaño funcional como la cuantificación de la dimensión de la funcionalidad del software, aspecto fundamental para establecer el tamaño final de un proyecto software. Recientemente se ha publicado la “ISO/IEC 20926: IFPUG Function Point Counting Practices”, necesaria para satisfacer los conceptos de medida de ISO/IEC 15504 y también para facilitar indicadores de progreso expresados en ISO/IEC 12207. La publicación de este nuevo estándar es resultado del proceso de normalización llevado a cabo por ISO/IEC e IFPUG (International Function Point Users Group), reflejando el esfuerzo de colaboración llevado a cabo por la industria y grupos de estandarización formal. La IFPUG es una organización no lucrativa, dirigida desde USA pero cuyos miembros representan a una gran variedad de empresas y organizaciones de desarrollo software distribuidas por todo el mundo.

Los IFPUG Puntos de Función miden el Tamaño “lógico” ó “funcional” de proyectos y aplicaciones software a partir de los requisitos funcionales de usuario (FUR, Functional User Requirements). Estos requisitos FUR, representan los procesos y procedimientos implementados por el software, independientemente de cómo operará el software y cómo será desarrollado. No incluyen, por tanto, requisitos de prestaciones ni de calidad del software.

Antes de describir el proceso de obtención de los IFPUG Puntos de Función, conviene detenerse un poco y analizar el significado de la palabra “usuario”. En el contexto de la medida de tamaño funcional, la palabra usuario significa cualquier persona, cosa, objeto, software, entidad, etc., fuera de los límites del software evaluado, que interactúe con dicho software. Esta definición de usuario es similar al término “actor” en metodologías orientadas a objeto, extendiendo su alcance a entidades que no tienen que ser necesariamente personas físicas.

El proceso de obtención de los IFPUG Puntos de Función comienza con un proceso de evaluación del software definido por sus correspondientes requisitos funcionales de usuario (FUR). Este proceso de evaluación consiste en la identificación de los denominados “Componentes Lógicos Base” (Base Logical Components, BLC).

La ISO/IEC 20926 evalúa el software estableciendo cinco (5) tipos de BLCs a partir de los Requisitos Funcionales de Usuario:

- **Internal Logical Files (ILF):** objetos ó grupos de datos mantenidos (creados, borrados, actualizados...) por el software evaluado.
- **External Interface Files (EIF):** objetos ó grupos de datos que existen en otro software y son referenciados por el software evaluado.
- **External Inputs (EI):** procesos cuya función es mantener uno ó más ILFs ó controlar el comportamiento de la aplicación.
- **External Outputs (EO):** procesos cuya función es presentar datos a un usuario e

incluyen, como parte del proceso, cálculos, datos derivados, actualizaciones de ILFs y / ó modificación del comportamiento de la aplicación. **External Queries (EQ):** procesos cuya función es presentar datos de ILFs/EIFs a un usuario y no incluyen, como parte del proceso, cálculos, datos derivados, actualizaciones de ILFs ni modificación del comportamiento de la aplicación.

La siguiente tabla muestra los valores de los IFPUG Puntos de Función asignados a cada BLC identificado, con sus distintas ponderaciones:

	LOW	AVERAGE	HIGH
ILF	7 FP	10 FP	15 FP
EIF	5 FP	7 FP	10 FP
EI	3 FP	4 FP	6 FP
EO	4 FP	5 FP	7 FP
EQ	3 FP	4 FP	6 FP

Tabla 1

Una vez los BLCs han sido identificados, evaluados y ponderados, el número de Puntos de Función obtenido es sumado y el número entero resultante corresponde al Tamaño Funcional del software evaluado.

UCP, Use Case Points (Puntos de Casos de Uso)

En 1993 el método de “Puntos de Casos de Uso” para estimación de tamaño en proyectos software orientado a objetos, fue desarrollado por Gustav Karner de la empresa Objectory, en la actualidad Rational Software. Este método es una extensión de los métodos “Function Point Analysis” y “Mk II Function Point Analysis” (adaptación del método FPA principalmente usada en Inglaterra). Fue presentado como una tesis doctoral en la Universidad de Linköping, basado en estudios realizados sobre proyectos software de pequeño tamaño. El trabajo es ahora propiedad de Rational Software.

La unidad de medida utilizada en este modelo, los Puntos de Casos de Uso, es utilizada también para cuantificar el Tamaño Funcional del software, partiendo de los Requisitos Funcionales de Usuario.

En este caso, el método es únicamente aplicable a metodologías de desarrollo de software orientado a objetos. Los Requisitos Funcionales de Usuario son reflejados en diagramas de Casos de Uso (Use Case Diagrams) en un primer análisis del software evaluado, utilizando la notación UML (Unified Modeling Language).

Los Puntos de Casos de Uso son obtenidos a partir del análisis de Casos de Uso del sistema, siguiendo los siguientes pasos:

1. Clasificación de Actores en los Casos de Uso atendiendo a la complejidad de la interacción con el software evaluado.
 - **Actores Simples (Simple):** entidad que interactúa con el software evaluado mediante una API (Application Programming Interface) definida. Peso asignado: 1.
 - **Actores Medios (Average):** entidad que interactúa con el software evaluado mediante un protocolo definido, como por ejemplo, TCP/IP. Peso asignado: 2
 - **Actores Complejos (Complex):** la entidad que interactúa con el software evaluado suele ser una persona mediante un GUI (Graphic User Interface) ó

página Web. Peso asignado: 3

El total de Actores Sin Ajustar (UAW) se calcula contando cuantos actores hay de cada tipo (por grado de complejidad), multiplicando cada total por su peso asignado, y sumando los productos.

UAW (Unadjusted Actor Weights) = Suma (actor * peso)

2. Clasificación de Casos de Uso atendiendo al número de transacciones presentes en la descripción de los Casos de Uso.

➤ **Casos de Uso Simples (Simple):** se realizan menos de 3 transacciones, se implementa con menos de 5 clases, ó utiliza una interfase simple con acceso a datos simples. Peso asignado: 5.

➤ **Casos de Uso Medios (Average):** se realizan entre 4 y 7 transacciones, se implementa con entre 5 y 10 clases, ó requiere más diseño de interfase con acceso a datos complejos. Peso asignado: 10

➤ **Casos de Uso Complejos (Complex):** se realizan más de 7 transacciones, se implementa con más de 10 clases, ó requiere complejas interfases de usuario con acceso a bases de datos complejas. Peso asignado: 15

Cada tipo de Caso de Uso es multiplicado por su peso correspondiente, y los productos son sumados para obtener el total de Casos de Uso Sin Ajustar.

UUCW (Unadjusted Use Case Weights) = Suma (Caso de Uso * peso)

Influencia de los Factores Técnicos: se calcula un Factor de Complejidad Técnica (TCF, Technical Complexity Factor), relacionado con el software evaluado y que influirá en la productividad.

En la siguiente tabla se recogen los 13 factores que se consideran incrementan la complejidad del software, indicando el peso asignado a cada uno.

Tfactor = Suma (ValorTx * Peso)

donde ValorTx tendrá los valores 0, 3 ó 5 dependiendo de la influencia de cada factor en el software evaluado.

TCF = 0,6 + (0,01 * Tfactor)

4. Influencia de los Factores de Entorno: en este caso se calcula

el Factor de Entorno (EF, Environmental Factor), relacionado, entre otros, con el nivel de experiencia disponible.

La siguiente tabla muestra los 8 factores que se tienen en cuenta para el cálculo de EF.

Factor	Descripción	Peso
T1	Sistema Distribuido	2
T2	Tiempos de Respuesta	2
T3	Eficiencia Usuario Final	1
T4	Procesamiento Complejo	1
T5	Código Reutilizable	1
T6	Fácil de Instalar	0,5
T7	Fácil de usar	0,5
T8	Portable	2
T9	Fácil de cambiar	1
T10	Concurrente	1
T11	Seguridad	1
T12	Acceso por terceras partes	1
T13	Entrenamiento especial para su uso	1

Tabla 1

Efactor = Suma (ValorFx * Peso)
 donde ValorFx tendrá los valores 0, 3 ó 5 dependiendo de la influencia de cada factor en el software evaluado.

$$EF = 1,4 + (-0,03 * Efactor)$$

5. Ajuste de los Puntos de Casos de Uso: una vez realizados los cálculos anteriores disponemos de UAW (Actores sin ajustar), UUCW (Casos de Uso sin ajustar), TCF (Factor de Complejidad Técnica) y EF (Factor de Entorno).

El número total de Puntos de Casos de Uso Sin Ajustar se obtiene sumando el número de Actores y Casos de Uso Sin Ajustar.

$$UUCP = UAW + UUCW$$

El tamaño funcional del software evaluado vendrá dado por el total de Puntos de Casos de Uso, ajustados por los factores previamente obtenidos.

$$UPC = UUCP * TCF * EF$$

Factor	Descripción	Peso
F1	Familiarizado con Proceso de Desarrollo	1,5
F2	Aplicación de la Experiencia	0,5
F3	Experiencia en Orientación a Objetos	1
F4	Capacidad de Análisis	0,5
F5	Motivación	1
F6	Requisitos Estables	2
F7	Trabajadores Tiempo Parcial	-1
F8	Dificultades con el Lenguaje de Programación	2

Tabla 3

SLOC, Source Lines Of Code (Líneas de Código Fuente)

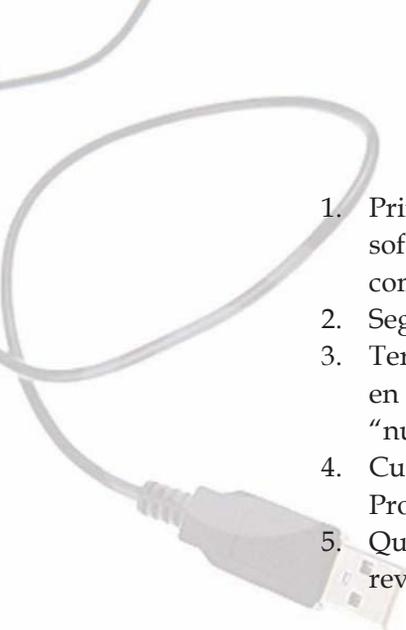
La métrica SLOC, también llamada LOC (Lines of Code), es una de las más conocidas y, sin duda alguna, utilizadas, para dimensionar el software. Utiliza un atributo interno e inherente al software, como es su número total de líneas lógicas, para establecer su tamaño. Proporciona información de forma inmediata cuando comparamos distintos proyectos, siempre que estos guarden una serie de similitudes.

Los proyectos software desarrollados en SAES suelen tener una serie de características comunes que los hacen apropiados para la utilización de esta métrica, como son la reutilización de código fuente; implementación de algoritmos de procesamiento de señal que obligan a la utilización de código secuencial (no OOD, Object Oriented Design); requisitos de tiempo real; y por último, generación automática de código adaptado a nuestras necesidades.

El método utilizado en SAES para la estimación del tamaño de proyectos software se basa en la utilización de esta métrica, partiendo de los siguientes puntos:

- Clasificación del software en tres (3) tipos:
 - Nuevo, más del 30% de cambios requeridos
 - Reutilizado, también llamado "As-Is", sólo requiere una adaptación de un 5%
 - Modificado, menos del 30% de cambios requeridos
- Banco de Datos de Productividad de SAES en diferentes proyectos en términos de Líneas de Código Fuente Lógicas, lenguaje de programación común y nivel de reutilización de código muy elevado
- Entornos de Desarrollo similares en los proyectos

El proceso seguido se describe a continuación:

- 
1. Primer Paso: descripción de las tareas a realizar dentro del ciclo de vida del software evaluado, por ejemplo, ingeniería de requisitos, diseño, pruebas, etc., así como de los productos a generar en cada fase (documentación, etc.)
 2. Segundo Paso: análisis preliminar del software a desarrollar
 3. Tercer Paso: código disponible para su reutilización. Estimación de SLOC basada en el análisis previo, módulos reutilizables y clasificación del SLOC estimado en “nuevo”, “As-Is” y “modificado”
 4. Cuarto Paso: aplicación del valor de productividad obtenido del Banco de Datos de Productividades, ajustado para este proyecto particular
 5. Quinto Paso: aplicación de factores de ajuste de riesgos técnicos, soporte para revisiones formales, asistencias establecidas, etc.

Conclusiones

El Tamaño Funcional nos proporciona una medida más objetiva del software, pero requiere un análisis de cierta profundidad que en ocasiones no es posible realizar.

Por otro lado, los Factores Técnicos y de Entorno juegan un papel fundamental en las Estimaciones de Coste. Los valores proporcionados por el Método Use Case Points no parecen muy ajustados tras aplicarlos en proyectos software de SAES y comparar los resultados estimados con los valores reales. Su ventaja frente a FPA es sin duda su orientación a objetos y la posible reutilización del análisis preliminar realizado en notación UML.

Sin embargo, para una cifra rápida e inmediata, y una estimación en un primer orden de magnitud, siempre persistirán las estimaciones de SLOC.

Fuentes y referencias

ISO BULLETIN May 2003. “Measuring the logical or functional size of software projects and software application”
(Carol A. Dekkers)

www.ifpug.org

www.softwaremetrics.com

“Use Case Points – An Estimation Approach” (Gautam Banerjee, August 2001)

<http://freeback.com/whitepapers/1035194512861.pdf>

<http://www.saspin.org/ProjectEstimateMethod.xls>

Evaluación de proveedores mediante la gestión de un repositorio de proyectos con una herramienta de estimación

ARTÍCULO

Evaluación de proveedores mediante la gestión de un repositorio de proyectos con una herramienta de estimación

La evaluación de ofertas de desarrollo de software, procedentes de diversos proveedores, es una tarea difícil para las organizaciones clientes. La variedad de las propuestas recibidas incluye las alternativas en las que el reducido coste es la característica más atractiva o las que prometen un optimista calendario. Sin embargo, en pocas ocasiones se estudia lo realista que pueda ser la oferta según la experiencia del proveedor o la calidad con la que se entregará el producto.

Introducción

Cada vez más, las organizaciones deciden subcontratar proyectos completos de desarrollo de software. Este tipo de proceso de compras requiere cierto nivel de madurez, tanto en la organización cliente, como en el proveedor.

Para mantener una relación cliente-proveedor justa, ambas organizaciones deberían haber cubierto todas las áreas clave del nivel 2 de CMM (1327) (Capability and Maturity Model): gestión de requisitos, planificación y seguimiento del proyecto, gestión de la configuración, aseguramiento de la calidad y gestión de la subcontratación.

La gestión de requisitos es fundamental para poder definir y controlar las características

contractuales que debe cumplir el producto y los cambios que se producen durante el desarrollo.

Los plazos de entrega especificados en el contrato deben estar determinados por una planificación realista, resultante de un adecuado proceso de estimación de proyectos. Se considera realista, cuando se basa en experiencias pasadas sobre proyectos de características similares y en condiciones similares.

Este plan debe ser la referencia sobre la que se establezca el seguimiento del progreso del proyecto y el control para corregir las desviaciones detectadas. Cada una de las entregas del producto deben ser controladas con los niveles de calidad establecidos en el contrato. Las versiones de los distintos componentes entregados se deben someter a lo establecido en los procedimientos de gestión de configuración establecidos.

De las áreas clave mencionadas, la gestión de requisitos y la planificación toman una relevancia especial a la hora de evaluar las ofertas procedentes de distintos proveedores. Suponiendo que todas las ofertas proporcionen la misma funcionalidad en su solución técnica, las condiciones propias del proveedor determinarán las diferencias entre las distintas planificaciones en las que se basa cada oferta.

Estimación de proyectos

Se ha mencionado la necesidad de utilizar las experiencias anteriores para llevar a cabo una planificación realista. Esta experiencia puede consistir en las vivencias que permiten a un

Autor Ramiro Carballo Gutiérrez

Publicado en la Revista Calidad – Julio-Agosto 2004

La evaluación de ofertas de desarrollo de software, procedentes de diversos proveedores, es una tarea difícil para las organizaciones clientes. La variedad de las propuestas recibidas incluye las alternativas en las que el reducido coste es la característica más atractiva o las que prometen un optimista calendario. Sin embargo, en pocas ocasiones se estudia lo realista que pueda ser la oferta según la experiencia del proveedor o la calidad con la que se entregará el producto.

Introducción

Cada vez más, las organizaciones deciden subcontratar proyectos completos de desarrollo de software. Este tipo de proceso de compras requiere cierto nivel de madurez, tanto en la organización cliente, como en el proveedor.

Para mantener una relación cliente-proveedor justa, ambas organizaciones deberían haber cubierto todas las áreas clave del nivel 2 de CMM [1][2][7] (Capability and Maturity Model): gestión de requisitos, planificación y seguimiento del proyecto, gestión de la configuración, aseguramiento de la calidad y gestión de la subcontratación.

La gestión de requisitos es fundamental para poder definir y controlar las características contractuales que debe cumplir el producto y los cambios que se producen durante el desarrollo.

Los plazos de entrega especificados en el contrato deben estar determinados por una planificación realista, resultante de un adecuado proceso de estimación de proyectos. Se considera realista, cuando se basa en experiencias pasadas sobre proyectos de características similares y en condiciones similares.

Este plan debe ser la referencia sobre la que se establezca el seguimiento del progreso del proyecto, y el control para corregir las desviaciones detectadas. Cada una de las entregas del producto deben ser controladas con los niveles de calidad establecidos en el contrato. Las versiones de los distintos componentes entregados se deben someter a lo establecido en los procedimientos de gestión de configuración establecidos.

De las áreas clave mencionadas, la gestión de requisitos y la planificación toman una

relevancia especial a la hora de evaluar las ofertas procedentes de distintos proveedores. Suponiendo que todas las ofertas proporcionan la misma funcionalidad en su solución técnica, las condiciones propias del proveedor determinarán las diferencias entre las distintas planificaciones en las que se base cada oferta.

Estimación de proyectos

Se ha mencionado la necesidad de utilizar las experiencias anteriores para llevar a cabo una planificación realista. Esta experiencia puede consistir en las vivencias que permiten a un experto generar una opinión sobre el nuevo proyecto; o, por el contrario, puede tratarse de un conjunto de datos numéricos que reflejan los resultados de proyectos históricos (repositorio de experiencias), que permiten aplicar técnicas de soporte a la decisión, para estimar el plan más probable.

Para construir el repositorio de experiencias, básicamente se debe disponer de la siguiente información por cada proyecto histórico:

- Tamaño funcional que resulte de medir la Especificación de Requisitos obtenida de la fase de Análisis, o bien, medida del tamaño del código fuente si el proyecto ha finalizado.
- Número de defectos detectados durante la fase de Pruebas.
- Esfuerzo realizado para desarrollar los requisitos medidos, que se obtendría de las tareas de Control del Proyecto (partes de horas, informes de seguimiento, etc.).

Con estas medidas base se obtienen dos métricas para cada proyecto:

- La Productividad: resultado de la relación Tamaño / Esfuerzo (Ej.: puntos función por mes hombre).
- La Tasa de Defectos: como la relación Defectos / Tamaño. (Ej. Núm. de defectos por punto función).

De esta manera, la estimación de un nuevo proyecto del que conocemos su tamaño funcional, permite conocer:

- El esfuerzo necesario = Tamaño / Productividad.
- El número de defectos esperados = Tasa de Defectos * Tamaño.

Hay que tener en cuenta que ésta es una visión simplista de la estimación de proyectos, ya que en la realidad, la productividad y la tasa de defectos se ven afectadas por multitud de factores dependientes del entorno, la organización, las características del equipo, el tipo de proyecto, etc.

El modelo de Putnam

En la búsqueda de un modelo para la estimación de proyectos en que se represente la realidad y la gran diversidad de factores del entorno que afectan al comportamiento de un proyecto, son conocidos los modelos de Boehm [3] (COCOMO, Constructive Cost Model), Putnam [4] [5] [6], etc.

En el caso de ambos modelos, se necesita disponer de un repositorio de proyectos históricos representativos de la organización que vaya a realizar el desarrollo.

El fundamento de estos modelos es el establecimiento de relaciones entre las distintas características registradas de cada proyecto. Por ejemplo, en el caso del Modelo de Putnam, el repositorio debe disponer de datos sobre:

- Esfuerzo realizado,
- Tamaño funcional,

- Duración y
- Defectos detectados.

Larry Putnam, utilizó un repositorio de proyectos de la Armada Americana para establecer la existencia de relaciones entre estos datos. Utilizando escalas logarítmicas en ambos ejes de una representación cartesiana, se advierte que el tamaño funcional medido en miles de ESLOC (Effective Source Lines Of Code) hace crecer la duración del proyecto medida en meses. Análogamente, se descubren las mismas relaciones del tamaño funcional frente al esfuerzo realizado y los defectos detectados. (Ver Fig.1).

En base a estas relaciones, el Modelo de Putnam, denominado SLIM (Software Lifecycle Management), se construyó sobre estas dos métricas:

El **Índice de Productividad (PI, Productivity Index)**, correspondiente al valor mapeado en una tabla que incluye un rango de resultados del Parámetro de Productividad, aplicando la siguiente fórmula:

$$\text{Parámetro de Productividad} = \text{Tamaño} / ((\text{Esfuerzo}/12)/B)^{1/3} * ((\text{Duración}/12)^{4/3})$$

- Donde B es un factor de ajuste
- Tamaño medido en ESLOC,
- Esfuerzo medido en Meses Hombre
- Duración medido en Meses

El **Índice de Presión de Tiempo (MBI, Manpower Buildup Index)**, indicador de la acumulación de esfuerzo en el tiempo para reducir la duración del proyecto, aplicando la siguiente fórmula:

$$\text{MBI} = \text{Esfuerzo} / \text{Duración}^3$$

Este indicador es una muestra de la dimensión de los equipos de desarrollo con los que la organización está acostumbrada a trabajar.

La Fig. 2 muestra las dependencias entre los factores clave del proceso de desarrollo.

El repositorio de Proyectos

Cuando se pretende evaluar las ofertas de diversos proveedores, la estimación debe basarse en indicadores de productividad asociados a un repositorio de proyectos propios de cada proveedor.

Si esto no es posible, se debe tener conocimiento de los niveles de productividad existentes en la industria del sector. Esto sólo se puede conseguir si

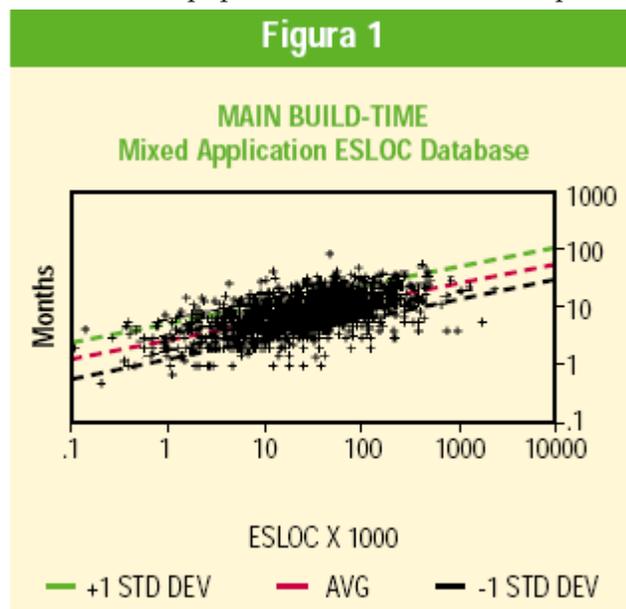


Figura 2

Los Factores Clave del Proceso de Desarrollo determinan Tiempo, Esfuerzo y Fiabilidad

Las relaciones entre ellos se deben cuantificar para ser útiles

Factores Clave	Medidas para la Gestión		
	Duración	Esfuerzo	Defectos
Tamaño	↑	↑	↑
Productividad	↓	↓	↓
Presión Tiempo	↓	↑	↑

El aumento del tamaño del proyecto supone el aumento de la duración del desarrollo, el esfuerzo dedicado y los defectos introducidos. La mejora en productividad supone una reducción de las tres medidas mencionadas. Sin embargo, la fuerte inversión en recursos con el objetivo de reducir la duración, supone una limitada reducción del tiempo de desarrollo, frente a un importante incremento de los costes asociados al esfuerzo realizado y una dramática reducción de la calidad del producto entregado.

se dispone de un repositorio en el que tengan cabida proyectos de múltiples organizaciones.

Para llevar a cabo las evaluaciones de ofertas que se describen a continuación, se ha utilizado la herramienta SLIM (Software Lifecycle Management) de QSM que implementa el Modelo de Putnam, y que incorpora un repositorio de 5.400 proyectos de diferentes clientes de QSM, clasificados, entre otros aspectos, por tipo de proyecto (Gestión, Telecomunicaciones, Tiempo Real, Web, etc.).

Se ha seleccionado esta herramienta por las facilidades que proporciona el interfaz gráfico, las posibilidades de almacenar y comparar distintas alternativas, y la inclusión del repositorio que proporciona información muy útil para conocer el comportamiento medio de la industria software.

Este repositorio se puede tomar como base para realizar estimaciones en organizaciones en las que se carece de experiencia registrada, para evaluar ofertas de proveedores respecto de la

media del mercado, o incluso para realizar comparaciones (benchmarking) de los procesos de desarrollo de una organización respecto de otras del sector al que pertenece.

El uso conjunto de la herramienta y el repositorio permiten realizar estimaciones en base a la productividad de proyectos anteriores, de tamaño funcional y tipo de aplicación similares al que se quiere estimar. La estimación resultante consta de la siguiente información:

- Esfuerzo total y por cada fase
- Distribución del esfuerzo en cada fase mediante curvas de Rayleigh
- Distribución de esfuerzo por categorías profesionales
- Número máximo de personas del equipo de desarrollo en cada fase
- Duración total y distribuida por fases
- Solapamiento entre fases
- Fechas estimadas de hitos predefinidos
- Defectos esperados en cada fase
- Tiempo medio entre defectos (MTTD) al final de cada fase
- Distribución de defectos en 5 niveles de criticidad

Figura 3

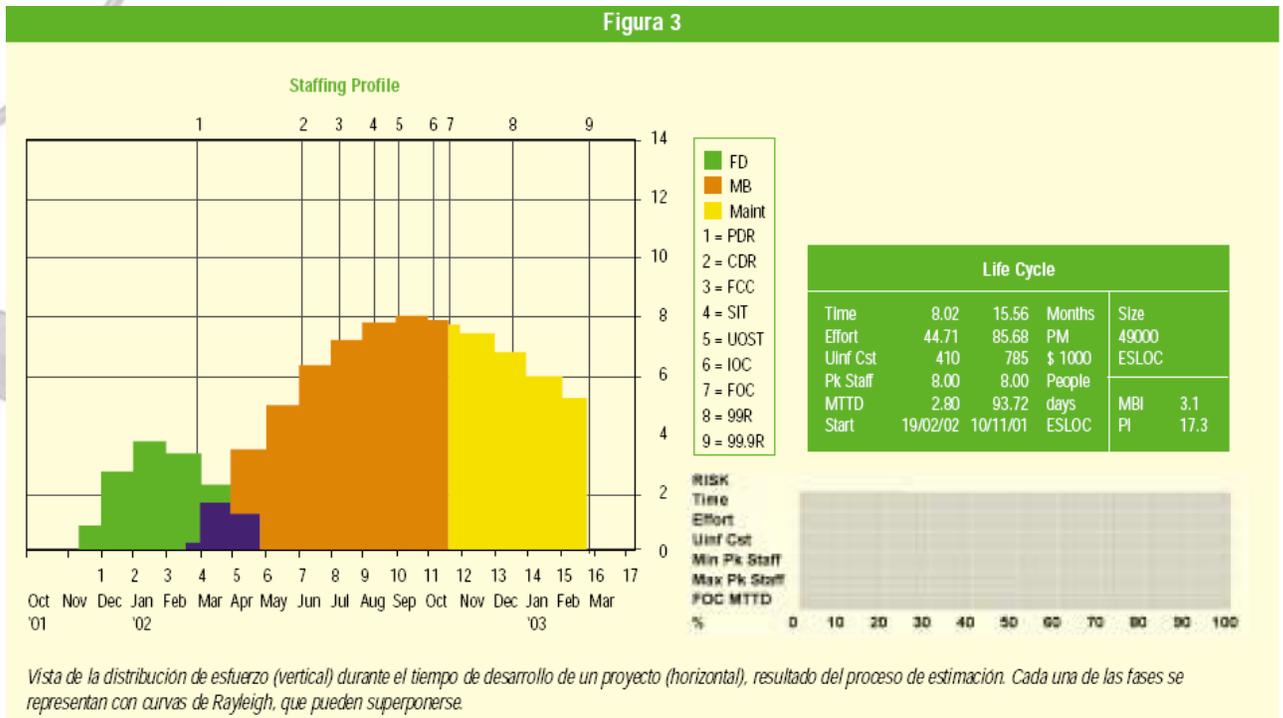
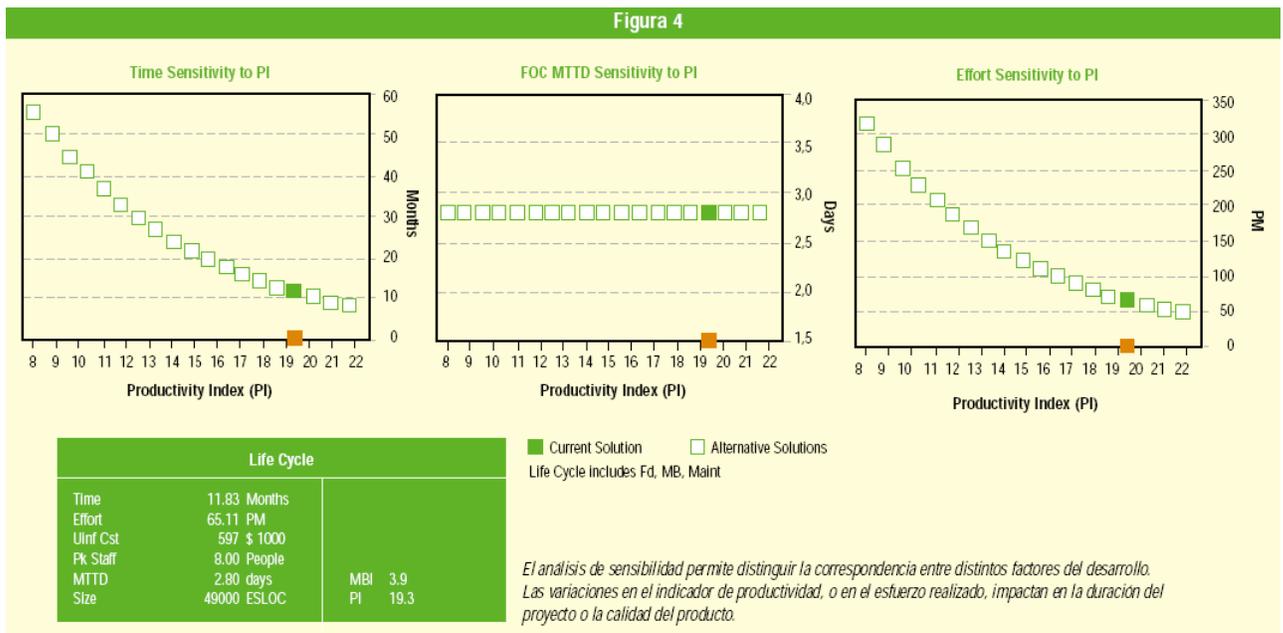


Figura 4



Subcontratación de un proyecto de desarrollo

Supongamos que una organización, que tiene un importante departamento interno de desarrollo de software, tiene todos sus recursos comprometidos a medio plazo, y decide la externalización de la construcción de un nuevo sistema.

La Dirección Corporativa ha determinado, por motivos de ventajas competitivas en su sector (arbitrariamente), que el nuevo sistema debe estar finalizado en un plazo de 12 meses.

Se ha recibido una oferta de un proveedor que se compromete a realizar el trabajo en 12

meses, con un equipo máximo de 8 personas, que se irán incorporando progresivamente al proyecto.

Aunque se dispone de una especificación funcional inicial, la ambigüedad de ésta lleva al proveedor a fijar un rango de incertidumbre entre 30.000 y 60.000 ESLOC de tamaño funcional estimado. El personal técnico de la empresa cliente, sugiere evaluar la oferta para un tamaño previsto de 51.000 ESLOC.

El Responsable de Desarrollo debe presentar un informe a la Dirección Corporativa, resultado de la evaluación de las alternativas propuestas por los proveedores. Para ello, utilizará los criterios del menor tiempo de desarrollo al menor coste posible, pero asegurando una garantía del 75% de probabilidad de cumplir el plan y presupuesto adjudicado.

Como parte de la oferta, el proveedor ha suministrado información acerca de su productividad en proyectos pasados.

El PI (Índice de Productividad) medio del proveedor es de 11, mientras que la productividad más alta ocurrida en un proyecto fue de 11,5. El MBI (presión de tiempo medio) es de 3,6.

El proceso de evaluación se inicia mediante la estimación del proyecto en base a la productividad media de la industria.

Proceso de estimación: según la productividad media de la industria

Se introducen las siguientes hipótesis de entrada en la herramienta de estimación:

- Tamaño Funcional: 51.000 ESLOC (con un rango de incertidumbre entre 30.000 y 60.000)
- Tipo de Proyecto: 20% Telecomunicaciones, 80% Gestión
- Índice de Productividad (PI): según el repositorio de SLIM, para este tipo de proyectos y este tamaño, el PI de la media de la industria es 17.3
- Equipo máximo de 8 personas.

La Figura 3 representa el primer plan estimado, destacando la siguiente información:

- Duración del Proyecto: 15,56 meses
- Esfuerzo necesario: 85,68 meses hombre
- Equipo de Trabajo: 8 personas
- Calidad (Tiempo Medio entre Defectos: MTTD) al final del proyecto: 93,72 días.
- Calidad (MTTD) al inicio de la fase de mantenimiento: 2,80 días.
- Presión de Tiempo (MBI): 3,1

Se archiva esta solución como la estimación base de la evaluación en la herramienta SLIM, con el identificador “PI de la Industria, 8 personas”.

Explorando alternativas

Partiendo de la estimación anterior, se utiliza la funcionalidad de Análisis de Sensibilidad que proporciona la herramienta, para descubrir el PI que necesitaría tener el proveedor para desarrollar el sistema en el tiempo contratado, es decir, 12 meses. (Ver Fig. 4).

Para ello, se mantienen los criterios de 8 personas como equipo máximo de desarrollo, y el mismo tamaño funcional y tipo de proyecto.

El Análisis de Sensibilidad muestra como se altera el tiempo de duración del proyecto y el esfuerzo necesario conforme se incrementa o reduce la productividad.

De esta manera, se descubre que el proveedor necesitaría tener una productividad de 19,3 para poder desarrollar el producto en 12 meses con 8 personas.

Si se comprueba la experiencia del proveedor en este tipo de proyectos, se advierte la gran

diferencia entre la productividad necesaria y la productividad real (11,0).

Se archiva esta solución como la estimación basada en “PI necesario, 8 personas”, con los siguientes resultados:

- Duración del Proyecto: 11,83 meses
- Esfuerzo necesario: 65,11 meses hombre
- Equipo de Trabajo: 8 personas
- Calidad (Tiempo Medio entre Defectos: MTTD) al final del proyecto: 93,72 días
- Calidad (MTTD) al inicio de la fase de mantenimiento: 2,80 días
- Presión de Tiempo (MBI): 3.9

Estimación basada en la productividad media del proveedor

Si se genera una nueva estimación, basada ahora en la productividad media del proveedor en este tipo de proyectos (11,0), se obtiene:

- Duración del Proyecto: 36,89 meses
- Esfuerzo necesario: 207,93 meses hombre
- Equipo de Trabajo: 8 personas
- Calidad (Tiempo Medio entre Defectos: MTTD) al final del proyecto: 93,72 días
- Calidad (MTTD) al inicio de la fase de mantenimiento: 2,80 días
- Presión de Tiempo (MBI): 0.5

Se archiva esta solución alternativa como la estimación basada en “PI medio de proveedor, 8 personas”.

Estimación basada en la presión de tiempo habitual

Buscando en la historia del proveedor, se encuentra que es habitual que los equipos de desarrollo trabajen con una presión de tiempo (MBI) en torno a 3,6 en media.

Pero si se analiza el detalle de cada proyecto del repositorio de experiencias del proveedor, se descubre una importante variación, con valores entre 1,7 y 5,4.

Si se mantiene la productividad media (PI) del proveedor (11) y la presión de tiempo media (MBI) en su equipo de trabajo (3.6). Para ello, se utiliza el interfaz gráfico de la herramienta, obligando a reducir el tiempo de desarrollo, lo que condiciona un fuerte incremento del número de miembros del equipo, y una disminución de la calidad entregada. Se obtiene la siguiente estimación:

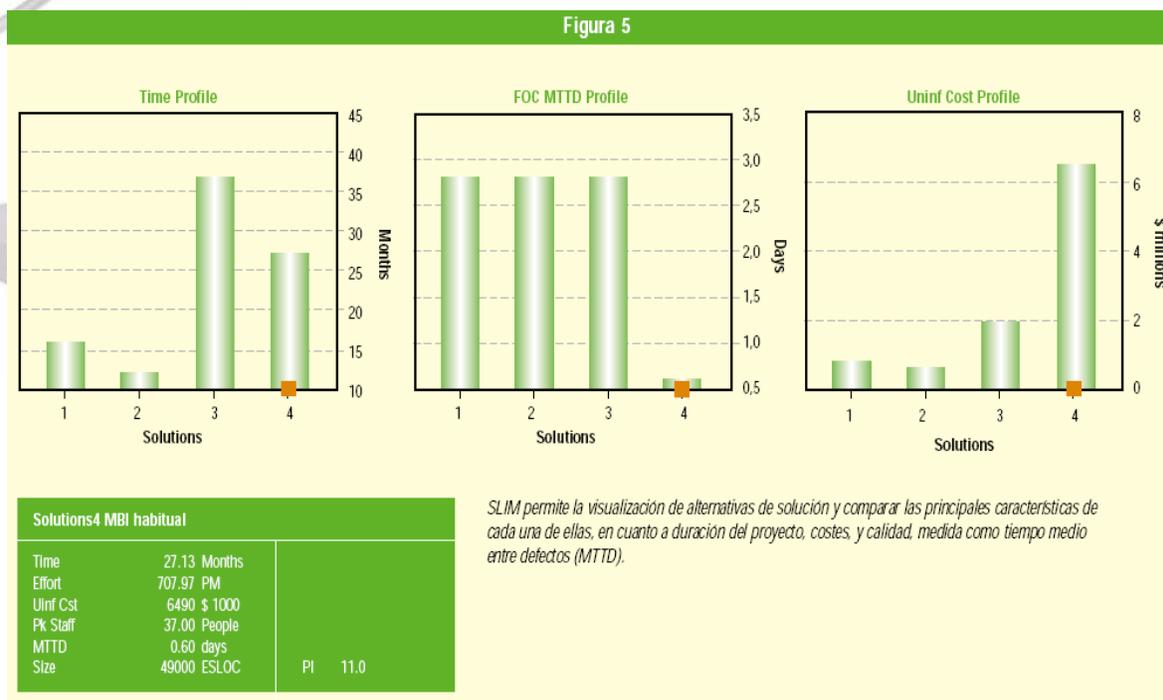
- Duración del Proyecto: 27,13 meses
- Esfuerzo necesario: 707,97 meses hombre
- Equipo de Trabajo Máximo: 37 personas
- Calidad (Tiempo Medio entre Defectos: MTTD) al final del proyecto: 20,26 días
- Calidad (MTTD) al inicio de la fase de mantenimiento: 0,60 días

Análisis de las alternativas

Mediante el registro de cada solución de estimación encontrada mediante los procesos antes descrito, la herramienta proporciona una visión global de las alternativas, en las que se advierten las relaciones entre duración, calidad de producto entregado y coste o esfuerzo.

En la primera estimación, quedó patente que la media de la industria de desarrollo de software tiene una productividad de 17,3 para este tipo de proyectos. Con esta productividad media, no es normal que un proveedor consiga realizar el desarrollo contratado con 8 personas

en tan sólo 12 meses. De momento, se hace notar que la Dirección Corporativa estableció la duración del proyecto de manera poco realista.



A continuación se busca un argumento para demostrar a la Dirección lo poco viable del calendario, mediante el cálculo de la productividad necesaria para cumplir los objetivos contratados: $PI = 19,3$.

Alcanzar un índice de productividad de este nivel puede suponer varios años de esfuerzo en mejora de proceso de desarrollo del software. Sin embargo, el ejemplo puede ilustrar el retorno de la inversión de un programa de mejora de proceso, ya que la diferencia entre tener $PI=19,3$ y tener 11,0 es de 140 meses hombre, para el mismo desarrollo.

Queda patente que no es habitual que el proveedor pueda implementar una productividad mejor que la indicada en sus proyectos históricos. Es decir, lo más probable es que el $PI = 11$ determine la forma de desarrollo. Pero si se utilizan 8 personas en el equipo de desarrollo, se triplica el tiempo de duración del proyecto fijado en el contrato.

Cuando se estudia la posibilidad de reproducir escenarios del pasado en los que el proveedor desarrolló con una presión de tiempo superior a la establecida por los 8 miembros del equipo, se descubre que se tuvieron experiencias equivalentes a que en este proyecto se incrementa el número de miembros hasta 37 personas.

Aún en este caso, que podría estar argumentado, el proveedor no podría cumplir los plazos de tiempo, y los criterios de calidad se verían seriamente perjudicados por el incremento del equipo de trabajo. Igualmente, el coste se hubiese multiplicado por 3,5.

De esta manera, se concluye que la propuesta analizada correspondiente a este proveedor, no es viable

Conclusiones

La utilización de herramientas de estimación de proyectos software como sistemas de ayuda

a la decisión que gestionan repositorios de proyectos, es un elemento fundamental para una evaluación de propuestas de proveedores, objetiva y realista.

Tradicionalmente, los responsables de adquisición de software de las empresas cliente se ven presionados por los financieros y otros directivos, para seleccionar aquellas ofertas que destacan por el ahorro en costes o la reducción de los plazos de entrega. En estos procesos, rara vez se considera la calidad del producto entregado o la viabilidad de realizar el desarrollo con los recursos propuestos.

En otras ocasiones, el deseo de anteponerse a la competencia en la puesta en marcha de nuevos sistemas, provoca el sobredimensionamiento de los recursos dedicados, con el objetivo de reducir el tiempo de entrega del producto.

Con las técnicas presentadas en esta ponencia, es posible contrastar la premura del tiempo con la calidad del producto entregado, basándose en el conocimiento de la productividad media de la industria de desarrollo de software, que proporciona el repositorio de proyectos.

Acrónimos

MBI: Manpower Buildup Index.

PI: Productivity Index.

MTTD: Mean Time To Defect.

Bibliografía y referencias

[1] Paulk, M.C., *Capability and Maturity Model for Software, Version 1.1*, CMU/SEI-93-TR-24, Technical Report, Software Engineering Institute, Carnegie Mellon University, 1993.

[2] Paulk, M.C., *Capability and Maturity Model for Software, Version 1.1*, CMU/SEI-93-TR-25, Technical Report, Software Engineering Institute, Carnegie Mellon University, 1993.

[3] Barry W. Boehm, *Software Engineering Economics*, Prentice Hall. 1.981.

[4] Lawrence H. Putnam y Ware Myers, *Measures for Excellence*, Prentice Hall. 1.992.

[5] Lawrence H. Putnam y Ware Myers, *Industrial Strength Software*, IEEE Computer Society Press. 1.997.

[6] Lawrence H. Putnam y Ware Myers, *Controlling Software Development*, IEEE Computer Society Press. 1.996.

[7] Kenneth M. Dymond, *Una Guía del CMM*, Process Inc. US. 1997



personas



procesos



tecnologías

El papel de la técnica de Puntos Función en el 'Outsourcing' del Desarrollo de Aplicaciones

Artículo

El papel de la técnica de Puntos Función en el 'Outsourcing' del Desarrollo de Aplicaciones



Todo está listo. Tras algunos meses de dura negociación se han firmado por fin los acuerdos. El contrato de outsourcing entre el cliente y el proveedor de desarrollo ha sido difícil pero se ha logrado: se han establecido tiempos de compromiso, mecanismos de revisión, indicadores de cumplimiento del servicio; se han incluido penalizaciones por retrasos e incluso alguna bonificación, se ha hecho un gran esfuerzo para dejarlo todo atado.

Día 1 de vida del acuerdo. Llega la primera petición. Es un nuevo desarrollo, un nuevo aplicativo para el área de Gestión de Patrimonios. El cliente, a través de los mecanismos acordados, documenta la nueva aplicación y la hace llegar al proveedor. Este último, tras analizarla, estima que el esfuerzo para el nuevo desarrollo es de 2.500 horas/hombre, para un trabajo de tres meses de duración y así se lo comunica al cliente.

El artículo Calidad 2009, publicado en el número 100 de la revista, es un artículo de la revista de la Asociación de Puntos de Función.

la hace llegar al proveedor. Este último, tras analizarla, estima que el esfuerzo para el nuevo desarrollo es de 2.500 horas/hombre, para un trabajo de tres meses de duración y así se lo comunica al cliente.

Día 2. Suena el teléfono de nuestro proveedor de desarrollo. El cliente, indignado, dice que el proveedor le ha pasado una estimación extremadamente alta, él no preveía un coste tan alto, no está de acuerdo con los cálculos realizados por el proveedor. Se convoca una reunión de urgencia para la tarde. Se encuentran el cliente y el proveedor. El uno justifica que el desarrollo debe costar menos diciendo que en trabajos similares realizados en el pasado se ha tardado mucho menos, el otro dice que la estimación es realista y que el cliente no tiene en cuenta todos los procesos que tienen que realizarse.

La pregunta que ambos tienen tras la reunión es "¿cuál es la parte del contrato de outsourcing que no se ha dejado cerrada y que ahora está provocando estos problemas?".

Este relato de ficción es más común de lo que les gustaría a los gestores de contratos de outsourcing en el área de desarrollo. Normalmente el punto clave en estos contratos de outsourcing es el siguiente: sin una técnica objetiva, asumida por ambas partes y contrastada para la estimación del tamaño de los proyectos, es muy difícil garantizar que tanto cliente como el proveedor estén satisfechos con la evolución del contrato de outsourcing.

Autor Sonia I. López Pérez

Publicado en la Revista Calidad – Marzo-Abril 2005

Todo está listo. Tras algunos meses de dura negociación se han firmado por fin los acuerdos. El contrato de outsourcing entre el cliente y el proveedor de desarrollo ha sido difícil pero se ha logrado: se han establecido tiempos de compromiso, mecanismos de revisión, indicadores de cumplimiento del servicio; se han incluido penalizaciones por retrasos e incluso alguna bonificación, se ha hecho un gran esfuerzo para dejarlo todo atado.

Día 1 de vida del acuerdo. Llega la primera petición. Es un nuevo desarrollo, un nuevo

aplicativo para el área de Gestión de Patrimonios. El cliente, a través de los mecanismos acordados, documenta la nueva aplicación y la hace llegar al proveedor. Este último, tras analizarla, estima que el esfuerzo para el nuevo desarrollo es de 2.500 horas/hombre, para un trabajo de tres meses de duración y así se lo comunica al cliente.

Día 2. Suena el teléfono de nuestro proveedor de desarrollo. El cliente, indignado, dice que el proveedor le ha pasado una estimación extremadamente alta, él no preveía un coste tan alto, no está de acuerdo con los cálculos realizados por el proveedor. Se convoca una reunión de urgencia para la tarde. Se encuentran el cliente y el proveedor. El uno justifica que el desarrollo debe costar menos diciendo que en trabajos similares realizados en el pasado se ha tardado mucho menos, el otro dice que la estimación es realista y que el cliente no tiene en cuenta todos los procesos que tienen que realizarse.

La pregunta que ambos tienen tras la reunión es "¿cuál es la parte del contrato de outsourcing que no se ha dejado cerrada y que ahora está provocando estos problemas?".

Este relato de ficción es más común de lo que les gustaría a los gestores de contratos de outsourcing en el área de desarrollo. Normalmente el punto clave en estos contratos de outsourcing es el siguiente: sin una técnica objetiva, asumida por ambas partes y contrastada para la estimación del tamaño de los proyectos, es muy difícil garantizar que tanto cliente como el proveedor estén satisfechos con la evolución del contrato de outsourcing. El cliente pensará que le están haciendo pagar la inocentada de no pedir ofertas a otros proveedores para contrastar costes y tiempos y el proveedor pensará que es esclavo de una relación en la que tiene unos precios negociados a la baja y, encima, el descontento permanente del cliente con las

estimaciones realizadas.

Puntos Función

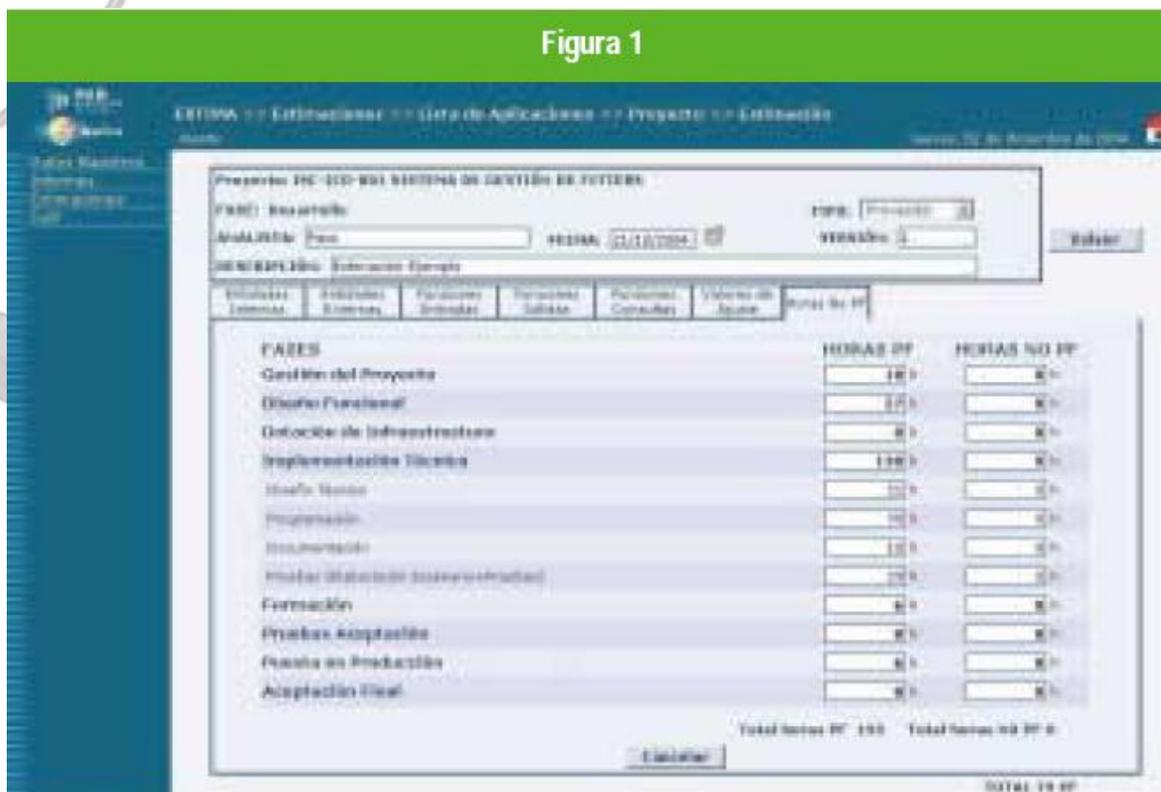
La técnica de Puntos Función permite la estimación objetiva del tamaño de un proyecto de desarrollo software basándose en la funcionalidad que el nuevo aplicativo proporciona al usuario y en la aplicación de unos factores de ajuste que dependen del rendimiento esperado, de la complejidad del proceso, etc. Esta técnica, aun sin ser la respuesta definitiva al problema, tiene una serie de características que la hacen adecuada para ser utilizada en los contratos de *outsourcing*:

- ***Perspectiva de usuario.*** La técnica obliga a realizar un esfuerzo en la mejora de la especificación de requisitos del aplicativo a desarrollar, eliminando la vaguedad y alejándose de detalles de implementación. Toda la especificación de requisitos deber realizarse con una perspectiva de usuario muy acentuada, lo que obliga al equipo de desarrollo a realizar un esfuerzo de comprensión de qué necesita el usuario y cómo lo percibe. Esta característica mejora la confianza entre el cliente y su proveedor, ya que se entienden perfectamente.
- ***Objetividad de la estimación realizada.*** Se acusa frecuentemente a Puntos Función de ser una técnica abierta en tanto que dos personas frente a un mismo problema estimarán dos tamaños distintos. Esto, aunque en general es así, puede paliarse con una formación específica de las personas implicadas en la estimación de los proyectos por parte del cliente y del proveedor, estableciendo un marco de referencia donde se indique qué es y qué no es cada uno de los elementos de la técnica. Si se cuenta con este marco de referencia la técnica es perfectamente objetiva al efecto de la estimación del tamaño de los proyectos. Además, las estimaciones de cliente y proveedor son comparables de forma que se pueden analizar los desencuentros y eliminarlos a través de un proceso negociador.
- ***Datos para aprender de la experiencia.*** La técnica de Puntos Función es en sí misma una extracción de métricas de cada nuevo desarrollo para estimar el tamaño del mismo. Esto nos lleva a coleccionar una información muy valiosa que puede ser analizada, enfrentando la productividad prevista, medida en horas previstas/puntos de función previstos, con la productividad real medida en horas reales/puntos de función reales. Estos valores permitirán a las empresas implicadas en un contrato de *outsourcing* ajustar los contratos y mejorar la satisfacción de cliente y proveedor.

Por otra parte la técnica de Puntos Función tiene situaciones en las que no recomendamos su utilización, como son las siguientes:

- La estimación de proyectos evolutivos en los que se eliminan funcionalidades para incorporar otras supone en algunos casos, sobre todo en aquellos en los que la complejidad es alta, que los puntos de función calculados no se vean afectados.
- Una estimación en puntos de función para proyectos muy pequeños puede tener un coste desmedido, en comparación con el tamaño del proyecto. Para ello hay que delimitar muy claramente qué proyectos vas a ser estimados con esta técnica y cuáles no.
- Por otra parte se conoce que el método en sí mismo no es eficaz en proyectos de gran complejidad lógica o en proyectos en tiempo real.

Figura 1



La experiencia: Proyecto para REE e INDRA

Tanto el Departamento de STI de RED Eléctrica Española como la Unidad de Outsourcing de INDRA, que mantienen un contrato de *outsourcing* de desarrollo de sistemas de información, desean evitar este marco tan común de incertidumbres ente cliente y suministrador. La orientación a la excelencia de ambas organizaciones les inclinan a incorporar a sus negociaciones anuales del contrato de *outsourcing* métodos y herramientas que incrementen cada vez más los niveles de claridad y transparencia en los que fundamentan su relación. La incorporación de un método objetivo de estimación del tamaño del software, con un enfoque cercano al usuario, comprensible y utilizable por ambas partes, y que no sólo aporte ese incremento en claridad y transparencia que se pretende, sino que también apoye la negociación entre ambos con una herramienta que les suministre información histórica como ayuda a la toma de decisiones, han sido las razones para pensar en Puntos de Función y en una herramienta software a medida que implemente la técnica de forma colaborativa. Entre los requisitos imprescindibles del proyecto que surgen a raíz de esta declaración de intenciones se destacan los dos siguientes; que tanto el método como la herramienta tienen que ajustarse a ambas organizaciones y a la metodología de trabajo ya establecida entre ambos, y no al contrario y que un tercero no involucrado fuera quien propusiera los mecanismos de estimación, los datos que nutrirían las negociaciones anuales del contrato de *outsourcing* y la realimentación del método. Ese rol fue asumido por Tecnología y Calidad de Software (TQS).

Fase I

La primera fase del proyecto se caracterizó por el asentamiento de las bases metodológicas ya existentes en REE e INDRA y porque los aspectos críticos relativos a la unicidad de lenguaje

quedarán asegurados.

Para la *unificación terminológica* se elaboró un glosario donde se tabulaban los términos habituales de REE/INDRA con los términos propios de método. Este glosario minimizaría el impacto de implantación eliminando la necesidad de adaptarse a nueva terminología. La unificación conceptual se logró mediante la elaboración de un Libro Blanco particular para la relación REE/INDRA que no sólo recogería la última versión del método de IFPUG2 (*ver 4.1.1. distribuida por AEMES3*) sino todas las recomendaciones necesarias para alcanzar la máxima objetividad posible del método, así como recomendaciones metodológicas relativas a la recogida de datos de los proyectos para nutrir una base de datos propia de la que realimentarse, orientada principalmente a las métricas de análisis de la productividad.

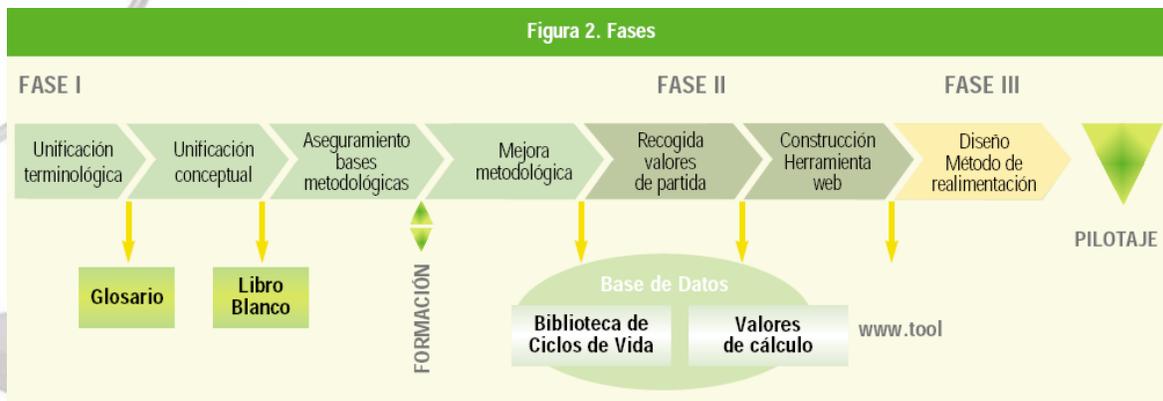
Como parte del aseguramiento de las bases metodológicas se impartió formación sobre la técnica de Puntos Función a todos los jefes de proyecto y analistas funcionales que finalmente tendrían que trabajar con la herramienta de soporte, tanto de REE como de INDRA. El resultado de este ejercicio práctico fue que tanto cliente como suministrador fueron capaces de identificar sin ambigüedades los elementos con los que trabajarían a partir de ahora (ILF's, EF's, etc.) y con un mínimo de adaptación semántica.

Es importante recordar que en el método de Puntos Función la valoración de los factores de ajuste en cierta medida perturba la posible objetividad de los puntos de función ajustados. Para paliar este riesgo se establecieron de común acuerdo los valores de los factores de ajuste estándar con los que se calcularían los PFA (Puntos Función Ajustados).

Estos factores de ajuste predeterminados nutrieron una biblioteca de lotes de factores, pudiendo ser modificados siempre que sea preciso pero bajo el acuerdo tanto del cliente como del suministrador, lo cual exige un razonamiento de dicha modificación y reduce el riesgo de falta de transparencia.

La segunda parte de la mejora metodológica se orientó a la *gestión del proyecto* que sería estimado en puntos de función y sus costes hora/hombre. Se estableció una *biblioteca de ciclos de vida* habituales para adaptarse a las distintas necesidades de los proyectos revisándose cada una de las fases y actividades y *determinando si dichas actividades se estimarían o no con el método*; y finalmente se estableció el porcentaje de dedicación sobre el total del proyecto necesario para cada una de las actividades.

Cada modelo de ciclo de vida recogió las actividades que se incorporaban al contrato de *outsourcing* y en las que se distribuiría el coste en horas del proyecto. Se identificaron datos de seguimiento de proyecto a recopilar así como de datos de cierre del proyecto. Estos datos, para cerrar la metodología, alimentarían las métricas necesarias para evaluar los proyectos finalizados. Dichas medidas, construidas tanto desde el punto de vista del cliente como del suministrador, ayudarían a ambas organizaciones no sólo a aprender de sí mismas sino que ayudarían en la toma de dediciones sobre proyectos futuros; es decir, servirían de base para las negociaciones anuales del contrato.



Con estos trabajos se ajustó el nuevo método de estimación y mejoró la metodología de gestión de proyectos ya existente sometiéndola a una revisión consensuada entre ambas partes al respecto de lo que incluía el contrato de *outsourcing* y lo que no, y lo que se iba a estimar en puntos de función y lo que no. Esta reorganización de la metodología de gestión de proyectos incrementó la conciencia del valor del seguimiento y cierre del proyecto.

Fase II

La segunda fase se compuso de dos grandes tareas, *la carga de los valores de partida y la construcción de una herramienta web a medida que recogiera el método de estimación.*

Para la alimentación de la base de datos histórica se procedió al análisis de los datos de un conjunto de proyectos de la base de datos del Repository For Release 8 de ISBSG4 (distribuida por AEMES3) la cual contiene información de más de 2.000 proyectos clasificados por tipología de proyecto (nueva aplicación, evolutivos) y tecnologías (Java, Power Builder, Visual Basic...) entre otras clasificaciones. Este conjunto de datos se sometió a refinamiento con los datos de los proyectos históricos de REE/INDRA de los cuales se realizó una estimación *post-mortem* y de los que, en base a su información de cierre de proyecto, se extrajeron otros datos importantes como la productividad por tecnología.

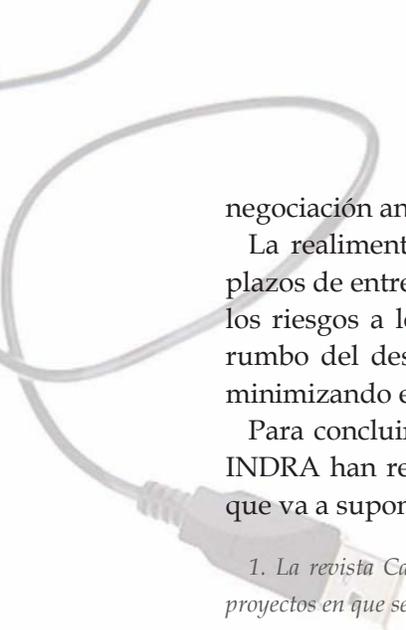
Una de las características que presenta la herramienta web que hemos desarrollado para dar soporte a la técnica de Puntos Función en la relación REE/INDRA es la imposibilidad, por ninguna de las partes, de alterar los valores de las variables del método sin someterlo a consenso.

La productividad aplicable, la biblioteca de ciclos de vida, incluso las propias estimaciones tienen una transparencia absoluta para cada una de las partes y son en todo momento sometidas a aprobación conjunta. La estimación no es sólo responsabilidad de una de las partes, son cliente y suministrador los que pueden estimar y valorar los proyectos y disertar sobre las diferencias. Una vez llegado a un acuerdo se almacena la estimación acordada.

En todo momento se pueden consultar las estimaciones históricas, tanto las de REE como las de INDRA y las que se acordaron finalmente para cada uno de los proyectos.

Fase III

Finalmente, en la tercera y última fase se diseñó el proceso de realimentación del método, la recogida de datos y el cálculo de las métricas. El servirse de un método como la estimación en puntos de función nos ayuda a tomarnos ese tiempo de cierre de proyecto como una nueva estrategia, nos anima a concluir el proyecto para obtener un beneficio cercano, el beneficio de alimentarnos de ellos, periodo tras periodo para ir ajustando los valores de nuestras variables (productividad, distribución de tiempo por fase, etc.) que serán las que sustenten la



negociación anual.

La realimentación permite el análisis de los factores que afectan a la productividad, a los plazos de entrega, a la calidad y al coste de los proyectos permitiendo a la organización prever los riesgos a los que está sometida y planificar su desarrollo (por ejemplo, puede afectar al rumbo del desarrollo de su metodología de proyectos o a su propia gestión de recursos) y minimizando el impacto de éstos.

Para concluir, se debe decir que, una vez pilotado el método desarrollado, tanto REE como INDRA han reconocido la practicidad de éste y la *notable mejora en las relaciones* entre ambos que va a suponer su implantación.

1. La revista *Calidad* marzo 2004, publicó un artículo sobre *Modelos para estimaciones del tamaño del software en proyectos en que se habla de la estimación en Puntos de Función*.
2. *International Function Point Users Group* (www.ifpug.org)
3. *Asociación Española de Métricas del Software* (www.aemes.org)
4. *International Software Benchmarking Standards Group* (www.isbsg.org)

Evolución de los planes de Métricas para el control de procesos de desarrollo de software



Autor: Ulises Arranz

Publicado en la Revista Calidad - Mayo 2005

Parece haber un generalizado consenso sobre el hecho de que una buena definición de procesos y su gestión con un enfoque de mejora continua son claves de la excelencia en el mundo del desarrollo y mantenimiento del software.

En la misma línea, se acepta y entiende que la gestión de dichos procesos en función de un conjunto de métricas definidas permite a la organización dirigir sus esfuerzos de mejora en el sentido en el que proporcione mejores rendimientos desde un punto de vista cuantitativo y cualitativo. La experiencia de las organizaciones involucradas en procesos de mejora continua apunta en esta dirección. Por lo tanto, uno de los pilares básicos del modelo es la correcta definición, recogida y análisis de métricas para el control de los procesos y su mejora.

Esto implica que las organizaciones que buscan la mejora continua debieran contar con un programa de métricas, no solamente dirigido a la gestión analítica de la producción, sino también a la cuantificación y control de los procesos de producción y de mejora continua.

Este artículo no es ni un manual de métricas, ni un curso de análisis estadístico, sólo pretende presentar la experiencia práctica de la evolución del Plan de Métricas dentro de una organización, la de los Centros de Desarrollo de Coritel en España, con sus procesos y productos controlados en base a indicadores. La evolución del Plan correrá paralela a la de la organización en la madurez de sus

procesos. Se explicarán los puntos principales de la definición e implantación de un Plan de Métricas y su aplicación para la gestión de los procesos de una organización.

La experiencia de Coritel en sus Centros de Desarrollo

Coritel, empresa perteneciente al Grupo Accentura, a través de sus Centros de Desarrollo en España (CSOC), tiene una experiencia única en nuestro país en la definición e implantación de procesos software. Es uno de pocas organizaciones en toda Europa que ha alcanzado el máximo nivel de madurez: el 5, en el modelo SW-CMM. En el modelo CMM, el nivel 4 es el que más profundamente incide en la recogida y análisis de métricas.

En la figura 1 se puede ver de forma esquemática la evolución vivida por CSOC en los sucesivos planes de métricas que ha aplicado.

Describiremos a continuación el significado y contenido de un Plan de Métricas, así como las características principales de cada uno de los planes que han sido aplicados a la organización, sus beneficios y las lecciones aprendidas.

Plan de Métricas

Un Plan de Métricas es el documento que describe en detalle la forma en que la organización define, aplica y analiza los indicadores importantes o relevantes para la misma.

Debe incluir el objetivo y propósito del plan así como el alcance definido, las políticas de confidencialidad, la definición de roles y responsabilidades para cada una de sus funciones, etc.

En el cuerpo principal del plan, se reubicaron de forma exhaustiva las características de cada indicador conteniendo el mismo:



[16] calidad >> mayo 2005

Este artículo no es ni un manual de métricas, ni un curso de análisis estadístico, sólo pretende presentar la experiencia práctica de la evolución del Plan de Métricas dentro de una organización, la de los Centros de Desarrollo de Coritel en España, con sus procesos y productos controlados en base a indicadores. La evolución del Plan correrá paralela a la de la organización en la madurez de sus procesos. Se explicarán los puntos principales de la definición e implantación de un Plan de Métricas y su aplicación para la gestión de los procesos de una organización.

Parece haber un generalizado consenso sobre el hecho de que una buena definición de procesos y su gestión con un enfoque de mejora continua son claves de la excelencia en el mundo del desarrollo y mantenimiento del software.

En la misma línea, se acepta y entiende que la gestión de dichos procesos en función de un conjunto de métricas definidas permite a la organización dirigir sus esfuerzos de mejora en el sentido en el que proporcione mejores rendimientos desde un punto de vista cuantitativo y cualitativo. La experiencia de las organizaciones involucradas en procesos de mejora continua apunta en esta dirección.

Por lo tanto, uno de los pilares básicos del modelo es la correcta definición, recogida y análisis de métricas para el control de los procesos y su mejora.

Esto implica que las organizaciones que buscan la mejora continua debieran contar con un programa de métricas, no solamente dirigido a la gestión analítica de la producción, sino también a la cuantificación y control de los procesos de producción y de mejora continua.

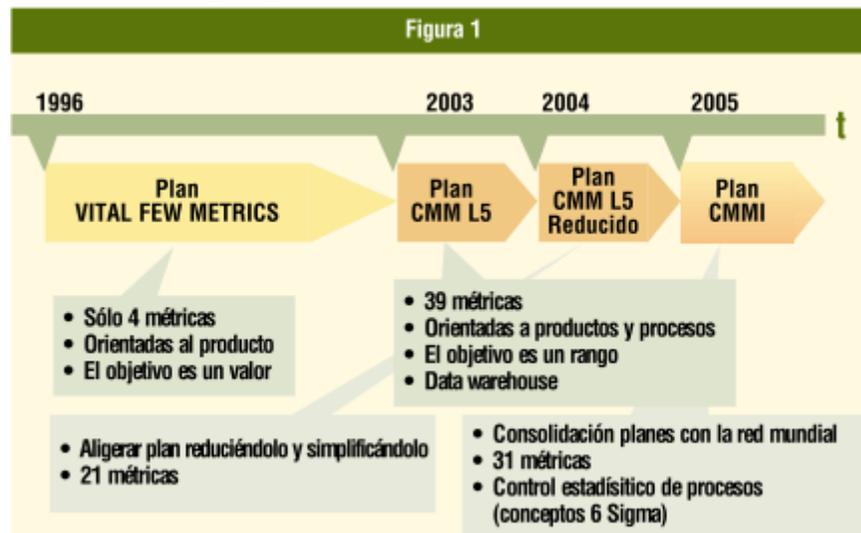
Este artículo no es ni un manual de métricas, ni un curso de análisis estadístico, sólo pretende presentar la experiencia práctica de la evolución del Plan de Métricas dentro de una organización, la de los Centros de Desarrollo de Coritel en España, con sus procesos y productos controlados en base a indicadores. La evolución del Plan correrá paralela a la de la organización en la madurez de sus procesos. Se explicarán los puntos principales de la definición e implantación de un Plan de Métricas y su aplicación para la gestión de los procesos de una organización.

La experiencia de Coritel en sus Centros de Desarrollo

Coritel, empresa perteneciente al Grupo Accenture, a través de sus Centros de Desarrollo en España (CSDC), tiene una experiencia única en nuestro país en la definición e implantación de procesos software. Es una de pocas organizaciones en toda Europa que ha alcanzado el máximo nivel de madurez, el 5, en el modelo SW-CMM. En el modelo CMM, el nivel 4 es el que más profundamente incide en la recogida y análisis de métricas.

En la figura 1 se puede ver de forma esquemática la evolución vivida por CSDC en los sucesivos planes de métricas que ha aplicado.

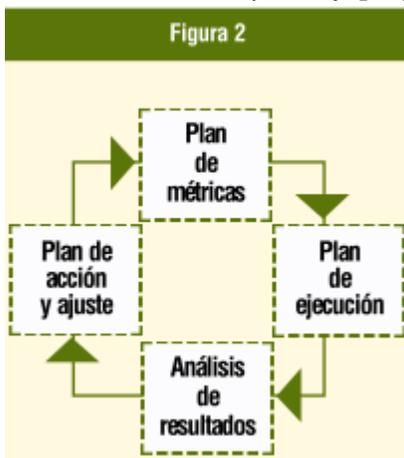
Describiremos a continuación el significado y contenido de un Plan de Métricas, así como las características principales de cada uno de los planes que han sido aplicados a la organización, sus beneficios y las lecciones aprendidas.



Plan de Métricas

Un Plan de Métricas es el documento que describe en detalle la forma en que la organización define, aplica y analiza los indicadores importantes o relevantes para la misma.

Debe incluir el objetivo y propósito del plan así como el alcance definido, las políticas de confidencialidad, la definición de roles y responsabilidades para cada una de sus funciones, etc.



En el cuerpo principal del plan, se relaciona de forma exhaustiva las características de cada indicador conteniendo al menos:

- Descripción/definición detallada de la métrica, su clasificación en un grupo de métricas dentro del plan, y su relación, si procede, con otras.
- Objetivo perseguido.
- Fórmula y cálculos a realizar con la definición de los operandos.
- Fuentes de información desde las que se obtendrán los operandos.
- Resultados cuantitativos y cualitativos esperados de la extracción de cada métrica.
- Aplicación en cada tecnología concreta, que podía sufrir modificaciones tanto en el sentido de la propia métrica como en la interpretación y cálculo (ejemplo: el cálculo de la productividad varía entre tecnologías).

- g) Otras informaciones de interés para la comprensión (ejemplos, referencias, comentarios, etc.).
- h) Rangos objetivos definidos por tecnología y tipología de trabajo para dicho indicador.

El Plan de Métricas es revisado de forma anual:

- a) Métricas a incluir.
- b) Rangos o niveles objetivos para cada indicador (partiendo de la información acumulada, la tendencia, los objetivos del anterior periodo, etc.).
- c) Operativas relacionadas con el análisis y la publicación de resultados.

Una vez revisado el plan, se publica, se imparte la formación oportuna necesaria y se implanta en los proyectos y procesos a los que va dirigido.

Las operativas han ido cambiando y evolucionando de un plan a otro, pero como norma general, los datos suelen ser recogidos de forma semanal desde las diferentes fuentes de información, tanto de interfaces desde las diferentes herramientas de gestión y de producción, como en algún caso vía introducción manual. La información obtenida se publica periódicamente. Los resultados son analizados a nivel de proyecto y de tecnología de forma periódica con un intervalo mínimo mensual. En las revisiones, el equipo de mejora continua analiza con los responsables de proyectos de cada tecnología, los resultados obtenidos en detalle, definiendo un plan de acción para el seguimiento de aquellos indicadores que deben ser mejorados.

Adicionalmente a los indicadores definidos y recogidos por el Plan de Métricas, existen otros que pueden ser definidos a nivel de proyecto para satisfacer las necesidades de información de cada uno de los proyectos, pero que no forman parte del plan. Éstos pueden ser indicadores internos o externos. Los internos nos ayudarán a controlar algún aspecto específico del desempeño en el proyecto, como por ejemplo el número de requerimientos bloqueados por dudas en la fase de diseño. Los externos son aquellos que acordamos proporcionar al cliente para facilitarle la gestión del proyecto, como pueden ser cuadros de mando para el seguimiento o indicadores de producto o proceso específicos. En este sentido son muy típicos los KPI (Key Performance Indicators) de los Acuerdos de Nivel de Servicio.

A continuación se describirán las características principales de los diferentes planes.

Plan de Métricas desde 1996 hasta 2003

El primer Plan de Métricas en CSDC (Vital Few Metrics) se remonta a 1996, fecha desde la que se comenzaron a recoger datos relativos a los productos software construidos, así como datos de gestión habituales a nivel de proyecto, pero que no eran consolidados o agregados para su análisis (estimaciones, desviaciones, etc.). Las métricas recogidas, para entornos mainframe, por entonces, se centraban en cuatro grandes conceptos:

Figura 3: Plan de Métricas desde 1996 hasta 2003

	Objetivo	Proporción	Puntos de medición	Categorías
CALIDAD	Garantizar que los productos entregados son correctos	Nº errores en los productos entregados /10.000 LOC	<ul style="list-style-type: none"> • Prueba de integración • Producción 	<ul style="list-style-type: none"> • Fallos • Mal funcionamiento
PRODUCTIVIDAD	Medir la capacidad de producción de ciertas tareas y permitimos realizar estimaciones más exactas en el futuro	Nº LOC día hombre	<ul style="list-style-type: none"> • A lo largo del proyecto 	<ul style="list-style-type: none"> • Centrado básicamente en tareas de construcción
FIABILIDAD	Medir nuestra capacidad para cumplir con las expectativas. Determinar la desviación entre los resultados esperados y los resultados conseguidos	<ul style="list-style-type: none"> • Diferencia de esfuerzo: nº horas est.-nº horas inc. / nº horas est. • Diferencia de calendario nº proyectos retrasados / nº total proyectos 	<ul style="list-style-type: none"> • A lo largo del proyecto 	<ul style="list-style-type: none"> • Fechas de entrega de cada producto
REUSABILIDAD	Medir la capacidad de optimizar el proceso de desarrollo reduciendo los errores del desarrollo manual	Nº LOC reutilizados / nº total LOC	<ul style="list-style-type: none"> • Al final de cada proyecto 	<ul style="list-style-type: none"> • Código reutilizado: plantillas, rutinas, arquitecturas, etc. • Código generado automáticamente por herramientas

Estas métricas se han mantenido en los siguientes planes definidos, como parámetros perfectamente implantados en la organización cuya evolución aporta una valiosa información histórica.

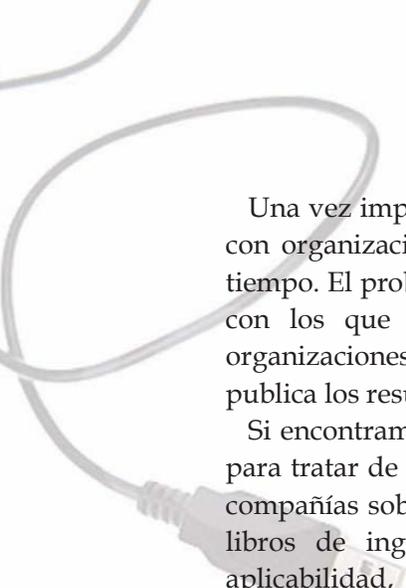
A pesar de la sencillez de los indicadores, permitían extraer información interesante sobre el impacto de la implantación de nuevas herramientas, mejoras en los procesos de ingeniería y sobre el resultado del producto. Por otra parte, la ausencia de datos sobre el propio proceso no permitía determinar con exactitud los mecanismos que originaban las eficiencias e ineficiencias en el mismo.

Este Plan de Métricas disponía de valores objetivos para cada uno de los indicadores definidos. La adherencia de los resultados a los valores objetivos se calculaba en bases a medias aritméticas y ponderadas.

Los beneficios y lecciones aprendidas de este longevo plan fueron claros. La selección de un conjunto de métricas pequeño y fácil de comprender, junto con disponer del tiempo suficiente para su estabilización y la activa publicación de los resultados a la organización fueron aspectos clave. Todo ello facilitó, por una parte, la institucionalización, y por otra, el grado de implicación y disciplina necesarios en las personas para la ejecución del plan.

La combinación de la simplificación sucesiva de la aplicación del plan vía herramientas, con la estabilidad en el tiempo y el análisis de los datos orientados a resultados concretos y prácticos en los proyectos, facilitó que las personas de la organización comenzaran a emplear las métricas como parte de su lenguaje de gestión, utilizando estos parámetros como criterios y argumentos de mejora del rendimiento y resultado objetivo de los proyectos.

La extracción de una estadística global por mes y una al cierre del año por métrica, permitía ir analizando las incidencias y discontinuidades producidas por distintos eventos acontecidos en un proyecto u organización en el tiempo, la relación o independencia existente entre algunas de las métricas entre sí y las tendencias. Era curioso observar el impacto en calidad por la incorporación de nuevo personal a un proyecto, o la mejora en productividad por la introducción de una nueva herramienta en el proceso de producción, o la relación entre productividad y reusabilidad, etc.



Una vez implantado y funcionando el Plan de Métricas, la tentación puede ser compararnos con organizaciones similares a la nuestra, hacer un benchmarking. Podemos perder mucho tiempo. El problema principal que encontraremos es la disponibilidad y fiabilidad de los datos con los que comparar. En muchos casos, nos encontraremos que la mayoría de las organizaciones no mide, o si mide, lo hace de diferente manera, o si lo hace de la misma, no publica los resultados de sus métricas, o peor, publica lo que quiere ser, no lo que es.

Si encontramos algunos datos apropiados, nos encontraremos haciendo engorrosos cálculos para tratar de comparar nuestros resultados con estudios publicados en Internet por distintas compañías sobre sus experiencias, algunos proyectos fin de carrera, estadísticas de anexos de libros de ingeniería del software y teoría de métricas, en muchos casos de dudosa aplicabilidad, realidad y utilidad. Nuestra conclusión, después de vivida esta experiencia, también fue que la única métrica fiable con la que te puedes comparar de verdad, y que te puede servir de referencia para el análisis de tu mejora y tendencia es la propia de periodos anteriores. De la misma forma, fijar una sana tensión en la definición de objetivos para el siguiente periodo empujará a la organización a mejorar y evolucionar.

Éstas fueron las principales lecciones aprendidas de la primera experiencia en la implantación de un Plan de Métricas en nuestra organización. No estábamos aún en la “tiranía de las métricas sobre los procesos”, pero se empezaban a sentar las bases. Todo ello fue creando la conciencia y las costumbres necesarias para la implantación de los planes de métricas siguientes, más complejos y extensos.

Plan de Métricas de 2003

Con la implantación de SW-CMM L4 y L5 en la organización durante ese año, fue necesaria la definición e implantación de un nuevo Plan de Métricas. Se deseaba mantener los indicadores del plan anterior y se necesitó añadir nuevos indicadores relativos a métricas orientadas a los procesos definidos de nivel 2, 3, 4 y 5, así como sus medidas de madurez y rendimiento, las medidas sobre el propio proceso de mejora continua definido y la ampliación de las métricas ligadas a producto.

Se amplió el número de datos a recoger (se paso de 4 a 39 indicadores) y se diversificaron las tecnologías objeto del programa (se extendieron a Netcentric y ERP's).

Asimismo, fue necesario cambiar el concepto de valor objetivo por el de rango objetivo. Se define un intervalo para cada métrica dentro del cual se consideran correctos los valores obtenidos por los diferentes proyectos y están sujetos a análisis cuando se sitúan fuera del mismo, superior o inferiormente, siendo necesaria la comprensión de las causas y la extracción de conclusiones para enfocar la mejora continua de los procesos.

Esta información es necesaria para los procesos de Nivel 5. La gestión de los procesos en función de las métricas obtenidas, enfocados, por lo tanto, a la mejora continua, se hace entonces posible.

En la figura 4 se incluye a modo de ejemplo la relación de métricas definidas en este segundo plan.

El incremento en el número de personas involucradas en el Plan de Métricas, así como el aumento en el volumen de información a recoger y analizar, unido a la necesidad de agregación de los datos, obligaron a la construcción e implantación de herramientas de recogida de información e interfaz, por una parte, y de análisis por otra. Entre las primeras se cuentan procesos de extracción, manipulación y carga de información desde herramientas de gestión y producción, y entre las segundas se cuenta con un DataWarehouse.

Disponer del DataWarehouse supuso la universalización de los indicadores definidos. La información era ahora susceptible de ser analizada con diferentes niveles de agregación. Desde el menor nivel posible, unidad de trabajo software, hasta el mayor, la organización en su conjunto, se puede obtener información de valor para cada uno de los diferentes niveles de responsabilidad del centro. La herramienta es empleada tanto por los jefes de proyecto, para el análisis de los resultados de su proyecto, como por los responsables de tecnología para medir la evolución en un determinado entorno, como por el responsable de centro para analizar cuál es la tendencia media, por ejemplo, de desviación en fechas de entrega.

Figura 4: Plan de Métricas de 2003

1. Productividad	21. Previsibilidad (a tiempo)
2. Efectividad (EED)	22. Previsibilidad (retraso)
3. Efectividad (Repetición del trabajo) / Nivel de repetición	23. Mejora de proceso estándar aceptada frente a recibida
4. Reutilización	24. Mejora de proceso estándar realizada frente a aceptada
5. Requisitos ok frente a total	25. Excepción a proceso estándar para procesar
6. Requisitos no ok frente a total	26. Excepción a proceso estándar para documentación interna
7. Requisitos devueltos frente a total	27. Excepción a proceso estándar para rangos
8. Requisitos con problemas frente a total	28. Excepción a proceso estándar para métricas
9. Problemas frente a requisitos	29. Cambios de configuración
10. Horas	30. Incumplimientos de SCM resueltos a tiempo
11. Diferencia de esfuerzo	31. Tiempo incurrido en la preparación del plan de SCM
12. Coste	32. Tiempo incurrido en el seguimiento de SCM
13. Calendario (%)	33. Incumplimiento de SQA resueltos a tiempo
14. Calendario (absoluto)	34. Tiempo incurrido en las revisiones de SQA
15. Horas recibidas frente a horas previstas	35. Tiempo incurrido en el seguimiento del plan de acción de SQA
16. Unidades recibidas frente a unidades previstas	36. Filtración de defectos/índice de error
17. Cambios de planificación	37. Diferencias de calendario
18. Tiempo medio para la resolución de modificaciones	38. Esfuerzo por componente
19. Tiempo medio de respuesta a problemas	39. Diferencias de tamaño
20. Tiempo medio de respuesta a aprobaciones	

Pero no todo es de color de rosa. El esfuerzo por analizar, definir y construir la herramienta no fue desdeñable y el principal problema encontrado en su explotación ha sido la validación de la información de entrada al sistema, que debe ser filtrada y necesita de intervención y revisión por parte de los responsables, debido a la heterogeneidad y variabilidad de las fuentes de información.

Este programa de métricas fue el primero que realmente combinaba indicadores de control sobre los productos y sobre los procesos. Su implantación supuso un fuerte soporte para la gestión de la organización y la mejora de los procesos.

Del análisis de los diferentes indicadores se empezaron a identificar aquellos que realmente aportaban información útil, respecto de aquellos que o bien eran redundantes o no proporcionaban valor real. Esta reflexión sentó las bases de lo que iba a ser la revisión del plan de cara al siguiente año.

Plan de Métricas de 2004

Bajo un modelo consolidado de CMM, esta revisión del Plan de Métricas del año se orientó, pues, a la definición del conjunto mínimo de indicadores y la constitución de un cuadro de mando de gestión. Este proceso de adelgazamiento mejoró el ratio coste/beneficio y racionalizó el Plan de Métricas. Para monitorizar este proceso y alinearlos con las necesidades reales, se creó un comité con representación de todas las tecnologías y del equipo de mejora continua, liderado por el responsable del Plan de Métricas. En diferentes sesiones se

seleccionaron los indicadores más significativos y se revisan y ajustan sus formulaciones. Se eliminan indicadores redundantes y aquellos que obtienen valores demasiado dispersos, lo que hace pensar que su definición no ha sido correctamente realizada. Se definen también nuevos indicadores.

En la figura 5 se enumeran las métricas que componen este plan (en color naranja, aquellas que han sido revisadas en su formulación).

Figura 5: Plan de Métricas de 2004

1. Productividad
2. Efectividad (EED)
3. Efectividad (Trabajo repetido) / Nivel de trabajo repetido
4. Reutilización
5. Horas
6. Diferencia de esfuerzo
7. Calendario (absoluto)
8. Horas recibidas frente a horas planificadas
9. Cambios de planificación
10. Tiempo medio de respuesta a problemas
11. Predictividad (retraso)
12. Mejora de proceso estándar aceptado frente a recibido
13. Mejora de proceso estándar realizado frente a aceptado
14. Excepción a proceso estándar para procesar
15. Excepción a proceso estándar para documentación interna
16. Excepción a proceso estándar para rangos
17. Excepción a proceso estándar para métricas
18. Incumplimientos SCM por revisión
19. Incumplimientos SQA por revisión
NUEVO Proporción de modificación...
NUEVO Índice de dudas...

El resto de las métricas, respecto al plan anterior han sido eliminadas. El resultado del plan está más alineado con las necesidades concretas de los procesos en la organización en ese momento.

Por supuesto, y a nivel de proyecto, como siempre, se siguen recogiendo y analizando los indicadores internos y externos necesarios. Estos indicadores no son cargados ni acumulados para su posterior análisis estadístico, pero siguen siendo vitales para el adecuado control de los proyectos.

A finales de este año Accenture define una estrategia global común en cuanto a modelo de procesos para

todos sus centros a nivel mundial (CMMI), así como la elaboración de un nuevo cuadro de mando basado en una nueva herramienta de recogida y análisis de métricas que pueda ser compartido por todos ellos. Estas iniciativas marcarán los cambios en el plan del año siguiente.

Plan de Métricas de 2005

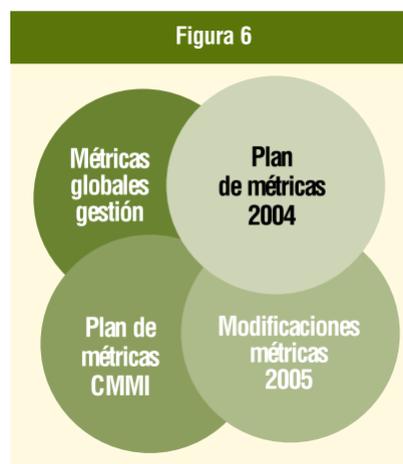
Puede este año ser considerado el de la globalización y alineamiento del modelo de procesos. Se toma la decisión estratégica de que todos los Centros de Desarrollo de Accenture a nivel mundial apliquen CMMI como modelo. Todos los centros principales

tienen planificado alcanzar Nivel 5 en el intervalo de los siguientes 24 meses. En el caso de CSDC, el objetivo es actualizar su SW-CMM Nivel 5 a CMMI Nivel 5 durante 2005.

Dentro de la misma iniciativa, y con objeto de homogeneizar la gestión y aumentar las sinergias, se busca estandarizar los planes de métricas de los diferentes centros (indicadores), así como el empleo de las mismas plantillas, informes, estadísticas, etc.

Adicionalmente, el resto de los proyectos Accenture compartirán un modelo común de cuadro de mando de indicadores para su gestión, alineado con prácticas igualmente CMMI.

Evidentemente este cuadro de mando de gestión ha de aplicarse igualmente en los Centros de Desarrollo, por lo que en nuestro caso, se han conjugado este año tal y como podemos



apreciar en la figura 6:

Sobre el plan de Métricas 2004 se han establecido las modificaciones y ajustes oportunos normales cada año, a los que se han sumado los nuevos indicadores necesarios por los cambios motivados por la evolución a CMMI y aquellas métricas de gestión que serán comunes a partir de ahora en todo proyecto a nivel mundial. Todo ello ha llevado de nuevo a una redefinición en el nuevo plan de la formulación y orientación de algunos de los indicadores y a la inclusión de bastantes otros.

Adicionalmente a esto, se está produciendo la migración desde la antigua herramienta de análisis de datos a la nueva.

El nuevo Plan de Métricas incluye los indicadores que se muestran en la figura 7.

Como se aprecia, algunas de las métricas han sido renombradas (e internamente reformuladas) o han cambiado su enfoque. Todo ello con objeto de mantener el mínimo número de indicadores claves y el máximo grado de alineación mencionado.

De igual forma, varias métricas pueden ser redundantes (por ejemplo Desviación en

Figura 7: Nuevo Plan de Métricas

MÉTRICAS ESPECÍFICAS CSDC		NUEVAS MÉTRICAS GLOBALES
1. Productividad		
2. Índice de errores (EED)		
3. Índice efectividad (reworking)		
4. Índice de reutilización de código		• Índice de rendimiento de costes
5. Volumen de dudas en requisitos		• Variación de costes
6. Volumen recibido vs volumen planificado		• Índice de eficiencia de la planificación
7. Tiempo medio de respuesta a dudas		• Variación de planificación
8. Ratio de modificaciones		• Variación a finalizar (Project VAC)
9. Desviación en horas		• Estimación estadística de variación a finalización (Statistical VAC)
10. Desviación en coste		• Índice de rendimiento a finalización (TCPI)
11. Desviación absoluta en plazo		• Ratio de riesgo realizado (RRR)
12. Entregas en plazo		• Ratio de exposición a riesgo (RER)
13. Nº mejoras aceptadas vs mejoras recibidas		• Contención de defectos de fase
14. Nº mejoras recibidas vs mejoras aceptadas		
15. Nº excepciones por proyecto al proceso estándar		
16. Nº excepciones por proyecto a los rango		
17. Nº excepciones por proyecto a las métricas		
18. Nº excepciones por proyecto a la documentación interna		
19. Nº medio de no conformidades de SCM por auditoría		
20. Nº medio de no conformidades de SQA por auditoría		
21. Métricas de satisfacción del cliente		

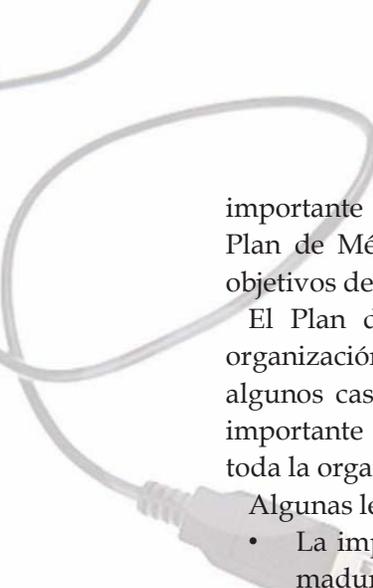
Coste vs Variación en Coste). Si se ha decidido mantener ambas durante este periodo es para permitir el análisis de los resultados de este año con los datos históricos acumulados en los años anteriores, mientras se obtiene información suficiente de los nuevos indicadores. En un futuro estas redundancias serán eliminadas.

Adicionalmente se ha comenzado este año con una definición de criterios de análisis estadístico basado en prácticas 6 Sigma sobre algunos indicadores del Plan, lo que permitirá la creación de modelos predictivos de resultados y unos criterios más rigurosos de revisión. Estos modelos se seguirán extendiendo en años sucesivos sobre los indicadores existentes y sobre otros nuevos que están previstos ser incluidos (por ejemplo, el ratio de defectos latentes, etc.).

En este Plan de Métricas se han definido conceptos nuevos para el análisis estadístico de los procesos, como son los límites de control (aplicados a nivel de proyecto y en función de su historia de desviaciones típicas), el rendimiento del proceso, cálculo de indicadores de capacidad del proceso (de conseguir su objetivo), herramientas para la detección de causas especiales, etc., cuya explicación excede el objeto de este artículo.

Conclusiones

He intentado explicar lo que un Plan de Métricas significa y el impacto que tiene sobre una organización y los procesos que la rigen. Seremos capaces de mejorar sólo aquello que podamos medir y tendremos que medir aquello que queramos mejorar, no todo. Es muy



importante una adecuada definición de objetivos para asegurar que existe una alineación del Plan de Métricas, así como de los procesos y de las prácticas de mejora continua con los objetivos de negocio de la organización.

El Plan de Métricas es un elemento en continua evolución y cambio dentro de una organización, que precisa de atención y seguimiento por parte de personas especializadas y en algunos casos con profundos conocimientos de estadística. Aparte de ello, necesita de una importante disciplina de trabajo y convencimiento por parte de los participantes en el Plan, toda la organización.

Algunas lecciones aprendidas extraídas de nuestra experiencia son:

- La implantación de un Plan de Métricas no es un proceso rápido, necesita tiempo para madurar, adecuarse a las necesidades de la organización y demostrar la validez y utilidad de sus resultados.
- No es práctico empezar por un plan muy ambicioso, con muchos indicadores. Es más eficaz comenzar con un conjunto de métricas pequeño, fácil de seguir, que nos permita definir acciones para mejorar los proyectos y luego hacerlo evolucionar.
- Es imprescindible involucrar a la organización entera en el proceso de definición e implantación del Plan de Métricas; desde los programadores hasta la dirección, cada uno a su nivel, debe entender el objetivo, y extraer beneficios de la información.
- Los resultados obtenidos, su análisis y los correspondientes planes de acción deben anunciarse periódicamente. Esto ayuda a entender a la organización el compromiso que existe con el Plan, además de poner a disposición de la misma la información.
- No se puede considerar los resultados obtenidos en un indicador como un valor absoluto. Su verdadero valor depende de las circunstancias que le rodean y de la relación con los procesos y otras métricas.
- Es preciso dotarse de herramientas que faciliten y agilicen la recogida, manipulación y análisis de la información.

El mayor beneficio que se puede obtener de mantener un Plan de Métricas es contar con información de la evolución de los procesos, sus resultados y tendencias. Con esta información se puede planificar un conjunto de acciones a corto, medio y largo plazo que nos permitirán mejorar nuestros proyectos y organización, verificando la eficacia de las acciones llevadas a cabo por medio de las propias métricas.

Cada Plan de Métricas dentro de CSDC ha supuesto un hito y una evolución en la madurez y en el control de los procesos. Al mismo tiempo que ha evolucionado la definición de procesos estándar de la organización, haciéndose más claros y completos, ha aumentado la necesidad de indicadores para medir la eficacia y comportamiento de los mismos.

A medida que la organización ha ido aumentando su grado de competencia en el empleo de métricas, éstas se han podido ir sofisticando, pudiéndose apreciar que su precisión ha corrido paralela con la madurez de los procesos a los que está monitorizando.

¡¡Ya no podemos vivir sin ellas!!



personas



procesos



tecnologías

Indicadores de Calidad del Software: el valor de los números.

Artículo

Indicadores de calidad

La implantación de indicadores de calidad en un proyecto de software se ha revelado como un elemento indispensable en la gestión del mismo. En este artículo se presenta la experiencia de GMV, S.A. (GMV) al respecto: se describe su motivación y su puesta en práctica en un proyecto real de envergadura y se discuten los costes y beneficios del mismo. Finalmente se proporciona una serie de recomendaciones acerca del conjunto mínimo de indicadores que, desde la experiencia de GMV y según el tipo de software, deben medirse para alcanzar un nivel de calidad satisfactorio.



Introducción

No se puede controlar lo que no se puede medir [De Marco, 1982].

Esta conocida afirmación, que podemos aplicar a cualquier aspecto de la vida, cobra cada vez más importancia en los procesos y actividades que se desarrollan dentro de cualquier organización.

Así, medimos los latidos del corazón para controlar que el ritmo cardíaco no supere ciertos límites que pongan en peligro nuestra seguridad; medimos los índices de polen en el aire para controlar ciertas afecciones respiratorias; tomamos medidas del número de litros caídos por cm³ para controlar posibles desbordamientos de ríos, por ejemplo; medimos, en definitiva, todo aquello que queremos controlar para ciertos fines.

Del mismo modo, medimos ciertas características de los procesos y actividades de una organización para entender, gestionar y

mejorar aspectos financieros, físicos, humanos, tecnológicos, de satisfacción del cliente/usuarios, etc. de dicha organización.

A la hora de medir, parece claro que lo primero que debemos hacer es definir una serie de indicadores basado en un conjunto de métricas.

Existen numerosas definiciones de lo que es un indicador. Según la norma [UNE 66175:2003], un indicador es un dato o conjunto de datos que ayudan a medir objetivamente la evolución de un proceso o de una actividad. Asimismo, una métrica es un criterio de medición según la misma norma. También define criterio de medición como acción y efecto de medir, de comparar una cantidad con su respectiva unidad, con el fin de averiguar cuántas veces la segunda está contenida en la primera.

La evolución de los indicadores a lo largo del tiempo permitirá comprobar, por tanto, la capacidad de los procesos para alcanzar los resultados planificados y ayudará en la toma de acciones de mejora.

[16] calidad >> junio 2004

Autores:

Raquel Rubalcaba Garrido

Almudena Sánchez González

Departamento de Calidad - GMV, S.A.

Publicado en la Revista Calidad - Junio 2004

La implantación de indicadores de calidad en un proyecto de software se ha revelado como un elemento indispensable en la gestión del mismo. En este artículo se presenta la experiencia de GMV, S.A. (GMV) al respecto: se describe su motivación y su puesta en práctica en un proyecto real de envergadura y se discuten los costes y beneficios del mismo. Finalmente se proporciona una serie de recomendaciones acerca del conjunto mínimo de indicadores que, desde la experiencia de GMV y según el tipo de

software, deben medirse para alcanzar un nivel de calidad satisfactorio.

Introducción

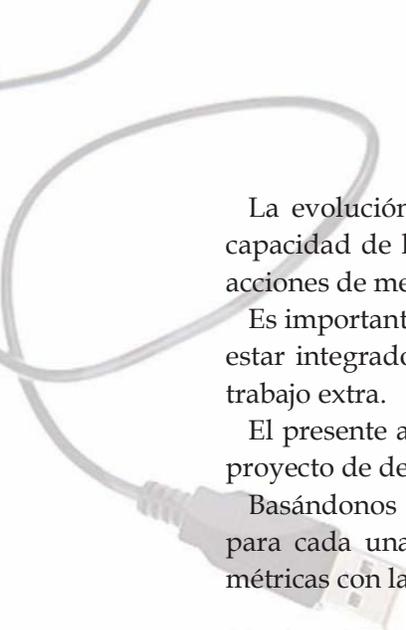
No se puede controlar lo que no se puede medir [De Marco, 1982]. Esta conocida afirmación, que podemos aplicar a cualquier aspecto de la vida, cobra cada vez más importancia en los procesos y actividades que se desarrollan dentro de cualquier organización.

Así, medimos los latidos del corazón para controlar que el ritmo cardíaco no supere ciertos límites que pongan en peligro nuestra seguridad; medimos los índices de polen en el aire para controlar ciertas afecciones respiratorias; tomamos medidas del número de litros caídos por cm³ para controlar posibles desbordamientos de ríos, por ejemplo; medimos, en definitiva, todo aquello que queremos controlar para ciertos fines.

Del mismo modo, medimos ciertas características de los procesos y actividades de una organización para entender, gestionar y mejorar aspectos financieros, físicos, humanos, tecnológicos, de satisfacción del cliente/usuarios, etc., de dicha organización.

A la hora de medir, parece claro que lo primero que debemos hacer es definir una serie de indicadores basado en un conjunto de métricas.

Existen numerosas definiciones de lo que es un indicador. Según la norma [UNE 66175:2003], un indicador es un dato o conjunto de datos que ayudan a medir objetivamente la evolución de un proceso o de una actividad. Asimismo, una métrica es un criterio de medición según la misma norma. También define criterio de medición como acción y efecto de medir, de comparar una cantidad con su respectiva unidad, con el fin de averiguar cuántas veces la segunda está contenida en la primera.



La evolución de los indicadores a lo largo del tiempo permitirá comprobar, por tanto, la capacidad de los procesos para alcanzar los resultados planificados y ayudará en la toma de acciones de mejora.

Es importante tener en cuenta que los indicadores seleccionados deberán ser fáciles de medir, estar integrados en la metodología de trabajo de la organización y no suponer una carga de trabajo extra.

El presente artículo se enfoca hacia la utilización de una serie de indicadores aplicados a un proyecto de desarrollo de software en GMV.

Basándonos en la experiencia pasaremos a presentar un caso real de un proyecto en el cual, para cada una de las fases del ciclo de vida del desarrollo de software, se eligieron unas métricas con las que se pudo medir las características de calidad del proceso y del producto.

Motivación y barreras

En el marco de trabajo de GMV, la normalización de sus procesos internos bajo [ISO 9001] así como su compatibilidad con modelos de excelencia y mejora continua hacen que la práctica habitual de sus desarrollos software planifiquen las actividades de calidad con objetivos medibles. La gestión cuantitativa de los procesos, particularizados para cada proyecto, permite conocer con precisión el progreso y alcance de la calidad requerida.

En concreto, el proyecto que nos ocupa requería especialmente por sus ajustados plazos de ejecución, por sus requisitos y por su ámbito de aplicación, un seguimiento objetivo a través de todas las fases del mismo. Requería un conjunto de indicadores enfocados, por una parte, al tamaño del producto y a su complejidad, y por otra, a la monitorización de los procesos.

Las principales barreras para la puesta en práctica del conjunto de indicadores elaborado eran, por un lado, su viabilidad mediante una recogida automatizada de datos y, por otro lado, la adopción por parte del equipo del proyecto de los umbrales definidos como los rangos aceptables para las características de calidad del producto o proceso.

En cuanto a la primera barrera anteriormente citada, el mercado ofrece herramientas que no siempre cubren todas las necesidades identificadas. Finalmente se optó por una solución mixta que comprendía herramientas comerciales, otras de libre distribución y también otras de desarrollo interno.

En cuanto a la segunda barrera, los umbrales que definen los rangos de aceptabilidad fueron considerados como objetivos a cumplir.

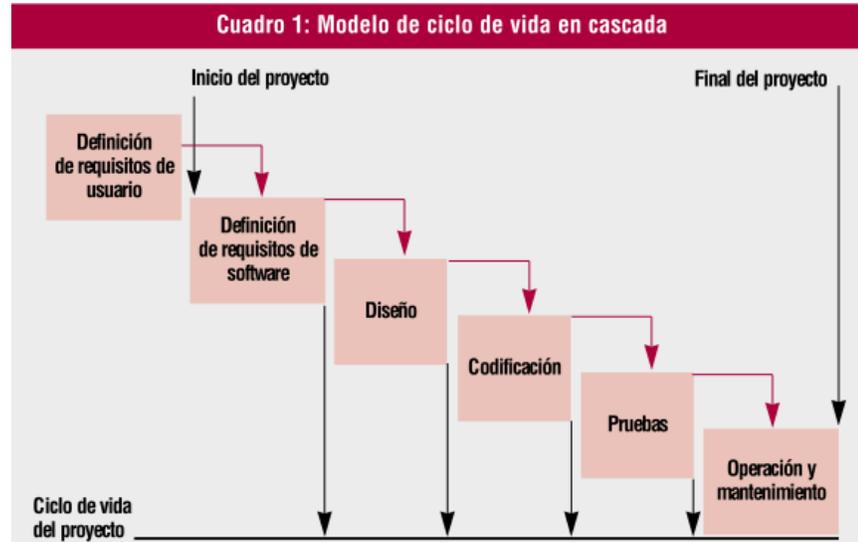
Así, en caso de desviaciones se analizaban las justificaciones documentadas y finalmente se decidía en base al sentido común y al juicio experto documentado de antemano en forma de heurísticas.

Caso real: proyecto de desarrollo Software

El proyecto utilizado para ilustrar la experiencia práctica del uso de indicadores de calidad de software recoge y mejora el conocimiento acumulado en GMV durante años en la materia. Entre los rasgos particulares del proyecto se podrían destacar que tuvo una duración de unos tres años de desarrollo con un equipo medio de unas quince personas y que tanto los requisitos de desarrollo de software como los de validación eran muy estrictos por la naturaleza o tipo de producto final. Este producto, desarrollado en código C, se encuentra actualmente en su fase de pre-operaciones y mantenimiento.

Siguiendo un modelo de ciclo de vida en cascada, las fases de desarrollo del proyecto son bien conocidas, y pueden representarse en el siguiente diagrama (ver cuadro 1).

Para cada una de las fases del ciclo de vida, se definieron una serie de indicadores que se detallan a continuación.



Indicadores seleccionados

La selección de indicadores se realizó de acuerdo a [ISO 9126]: se identificaron características

Tabla 1: Métricas seleccionadas

Métrica	Fase	Requisitos	Diseño	Codificación	Pruebas	Pre-operaciones y Mantenimiento
Tamaño del producto						✓
Líneas por función o clase				✓	✓	✓
Líneas del código				✓	✓	✓
Complejidad ciclomática				✓	✓	✓
Frecuencia de comentarios				✓	✓	✓
Número de niveles de anidamiento				✓	✓	✓
Número de puntos de salida				✓	✓	✓
Número de discrepancias y problemas		✓	✓	✓	✓	✓
Estabilidad de requisitos			✓	✓	✓	✓
Cobertura de sentencias				✓	✓	✓
Progreso de pruebas					✓	✓
Deslizamiento de hitos en calendario		✓	✓	✓	✓	✓
Número de acciones		✓	✓	✓	✓	✓
Número de riesgos		✓	✓	✓	✓	✓

a evaluar, se especificaron sus subcaracterísticas y, por último, se definieron las métricas y sus rangos de valores aceptables. Se dio preferencia a una expresión cuantitativa de las métricas frente a una cualitativa.

Las características a evaluar fueron: la funcionalidad del producto, su fiabilidad y su mantenibilidad. En el Apéndice 1 se encuentra la asociación entre estas

características y las métricas de la tabla que sigue a continuación.

Dichas métricas se presentan con una marca que muestra su aplicabilidad en la fase correspondiente del ciclo de vida (ver tabla 1).

Discusión

Con la perspectiva que nos otorga el tiempo de aplicación del programa de métricas arriba expuesto y sus indicadores asociados, podemos realizar un análisis crítico del mismo. La discusión que prosigue recopila las lecciones aprendidas de esta experiencia, diferenciando entre los indicadores asociados a fases concretas del ciclo de vida de aquellos independientes

del mismo.

En general, podemos avalar la utilidad práctica de medir el producto y los procesos del desarrollo de software, lo cual ha permitido una toma de decisiones objetiva y a tiempo.

Los indicadores independientes de las fases del ciclo de vida han sido útiles para la identificación temprana de riesgos cuando se analizaba su tendencia en el tiempo.

Por ejemplo, si representamos el indicador del seguimiento de acciones del proyecto con un gráfico de áreas, podemos observar cómo las acciones van creciendo a lo largo de la vida del proyecto y cómo se van resolviendo progresivamente. El momento actual representa las acciones pendientes de ser cerradas (ver cuadro 2).

El mismo tratamiento se utilizó para el seguimiento de problemas a lo largo del tiempo.



En cuanto a los indicadores asociados a fases concretas veamos los más significativos.

La aportación de la estabilidad de

requisitos ha sido esencial para visualizar e identificar de manera temprana muchos problemas de calendario, pero no todos.

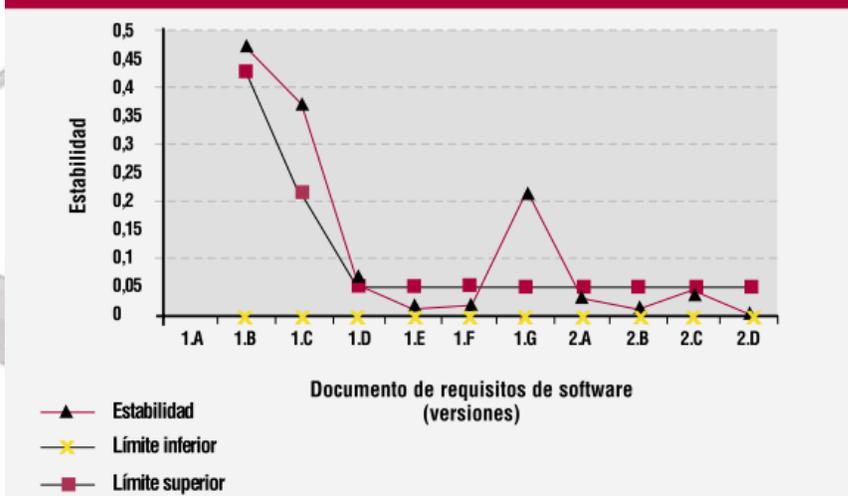
Tabla 2

Documento de requisitos de software (versiones)	Total requisitos	Añadidos/modificados/eliminados	Estabilidad	Límite inferior	Límite superior	% cambios
1.A	925	0	-	-	-	-
1.B	998	435	0,44	0,00	0,4	44%
1.C	1.715	605	0,35	0,00	0,2	35%
1.D	1.748	109	0,06	0,00	0,05	6%
1.E	1.755	17	0,01	0,00	0,05	1%
1.F	1.749	28	0,02	0,00	0,05	2%
1.G	1.766	348	0,20	0,00	0,05	20%
2.A	1.772	55	0,03	0,00	0,05	3%
2.B	1.787	21	0,012	0,00	0,05	1%
2.C	1.812	64	0,04	0,00	0,05	4%
2.D	1.814	5	0,003	0,00	0,05	0%

Las causas de la inestabilidad de los requisitos han venido de diversas fuentes. Por un lado, originadas a partir de los requisitos del usuario en una fase temprana del ciclo de vida, y por otro lado como consecuencia de vacíos de especificación identificados sólo durante el proceso de validación. En el

primer caso, analizar la estabilidad de requisitos ha resultado de gran utilidad. En el segundo caso, los problemas de calendario no se pudieron anticipar: ante un problema surgido en el proceso de validación, se efectuaba un análisis para resolverlo en el código fuente, trazar al diseño y por último a los requisitos de software.

Cuadro 3: Estabilidad de requisitos



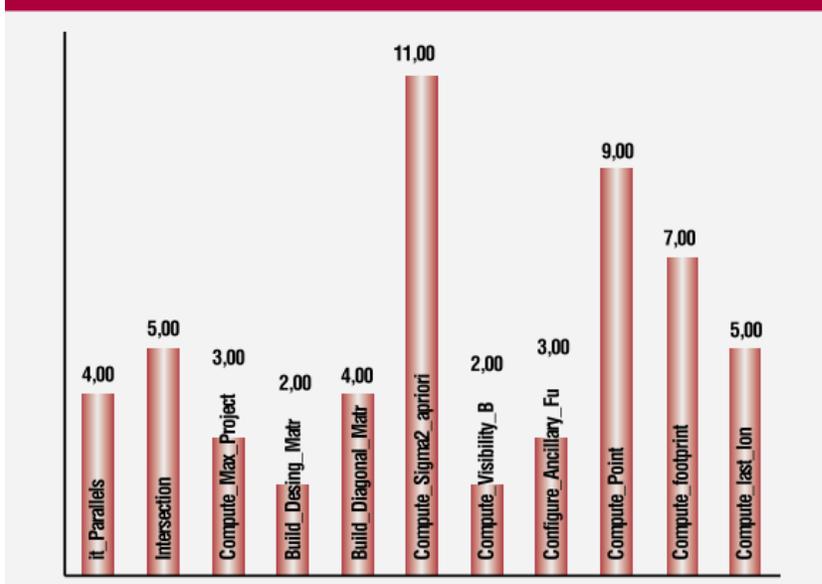
Los datos obtenidos para la estabilidad de los requisitos que se reflejan en la tabla 2.

Representando el indicador de la estabilidad de requisitos mediante un gráfico de líneas, podemos observar cómo los valores quedan dentro de los umbrales definidos y

cuando se produce una desviación (ver cuadro 3) se toman las acciones para corregirlo, hasta conseguir que se mantenga dentro de los límites.

Durante la fase de codificación, mantener un seguimiento estricto de la cantidad de líneas totales de código, líneas por función, la cantidad de comentarios y de la complejidad ciclométrica ha facilitado la depuración del código, así como las tareas propias de su mantenimiento.

Cuadro 4



El indicador que mayor esfuerzo requirió mantener dentro de los umbrales establecidos fue el de la complejidad ciclométrica debido en gran medida a la naturaleza compleja de los algoritmos a implementar.

Esta métrica desarrollada por McCabe proporciona una medida cuantitativa correlada con la dificultad para probar un módulo, esto es, cuando la complejidad

ciclométrica de un módulo supera un cierto umbral, resulta extremadamente difícil probarlo. Este indicador está muy relacionado con el número de niveles de anidamiento y también con el tamaño del módulo o función.

En el cuadro 4, representamos los resultados de la medición con histogramas donde se muestra el valor de la complejidad ciclométrica para cada una de las funciones que componen un módulo:

El número de niveles de anidamiento y el número de puntos de salida por función no han sido métricas muy relevantes, simplemente se han utilizado de modo complementario (la

primera, ya que un alto número de niveles de anidamiento incrementaba la complejidad ciclomática) o auxiliar (la segunda, utilizada para facilitar la traza del flujo de ejecución).

El indicador de cobertura de sentencias ha sido clave para determinar la finalización del nivel de verificación deseado y, por tanto, esencial para calcular el progreso en las fases de pruebas.

Se realizaron campañas de pruebas unitarias para asegurar que las sentencias de código fuente se ejecutaban en el porcentaje marcado en el umbral, también con pruebas de integración y de sistema alcanzando los umbrales previamente establecidos.

Entre otras lecciones aprendidas se encuentran las siguientes:

- Equilibrar la cantidad de indicadores; esto es, reducir quizá la cantidad de indicadores durante la fase de codificación manteniendo sólo aquellos que realmente han sido los conductores de las decisiones.
- Establecer los indicadores más singulares de cada fase; es decir, no parece lógico a la vista de los resultados utilizar los mismos indicadores por ejemplo, en pruebas y en codificación.
- Existen otros indicadores relevantes que se podrían utilizar ya que proporcionan información complementaria, tales como el progreso de pruebas ejecutadas frente a planificadas.

Conclusiones

La experiencia aplicada de establecer un programa de métricas en un proyecto de desarrollo de software ha demostrado que los indicadores de calidad son una herramienta imprescindible para una gestión eficaz y eficiente de los procesos. La evaluación de la calidad deja de ser una tarea intuitiva para tomar un cariz objetivo y analítico.

Entre los beneficios obtenidos de su utilización cabe destacar la pronta identificación de riesgos gracias al análisis de tendencias de los indicadores seleccionados. Esto permite la minimización de problemas y el incremento de la satisfacción del cliente.

Además, el traducir el concepto abstracto de calidad en términos medibles ha favorecido la toma de conciencia del personal hacia los atributos de calidad. Asimismo, ha facilitado los procesos de verificación y de mantenimiento del producto.

Por todo esto, nuestra experiencia nos permite recomendar en los proyectos de software el uso de indicadores a partir de un programa de métricas. Dicho programa debe estar ajustado por una parte al proyecto, esto es, debe ser coherente con los objetivos específicos del mismo, y por otra, al nivel de madurez de los procesos en la organización y en el propio proyecto.

No obstante, recomendamos como esenciales métricas tales como estabilidad de requisitos, número (y tendencia) de problemas y de acciones, líneas totales de código producido, complejidad ciclomática, líneas por función (o método) y por clase (o módulo) y cobertura de sentencias.

Hay que considerar que la validez de las métricas apuntadas podría dejar de tener sentido en tipos de aplicaciones o entornos de desarrollo y verificación muy diferentes al descrito.

Para concluir, resaltamos dos cuestiones fundamentales desde el punto de vista de la organización:

1. Disponer unos indicadores bien definidos permite eliminar las posibles ambigüedades y distintas interpretaciones; permite comparar los proyectos de manera objetiva; se pueden predecir costes, tiempos y calidad de los proyectos; se pueden corregir errores de manera temprana, es posible adoptar compromisos de manera eficiente y se pueden justificar

decisiones al disponer de los datos objetivos.

2. La adopción de indicadores de calidad en los proyectos deberá institucionalizarse de manera paulatina. La adopción de las prácticas de medición en los desarrollos de software irá alimentando una base de datos de la organización de importancia mayúscula para la puesta en marcha de programas de mejora continua.

Referencias

[Tom de Marco]. *“Controlling Software Projects: Management, Measurement and Estimation”*. Englewood Cliffs, NJ: Yourdon Press, cop. 1982.

[UNE 66175:2003] *“Sistemas de Gestión de la Calidad. Guía para la implantación de sistemas de indicadores”* AENOR 2003.

[CMU/SEI-93-TR-024] *Capability Maturity ModelSM for Software, Version 1.1, February 1993.*

[ISO 9126] *Information technology -Software product evaluation -Quality characteristics and guidelines.*

[ISO 12207] *Industry implementation of International Standard ISO/IEC 12207:1995 - Software life cycle.*

[ISO 9001:2000] *Quality management systems. Requirements.*

[IEEE Std 1061-1998] *IEEE Standard for a Software Quality Metrics Methodology.*

Apéndice 2: Definición de los indicadores seleccionados

→ Tamaño del producto

Se mide en términos de múltiples atributos que se recogen normalmente al final del proyecto y que sirven de lecciones aprendidas para su utilización en otros proyectos

- Número de requisitos de usuario
- Número de requisitos de software
- Número de componentes de diseño de alto nivel
- Número de componentes de diseño detallado
- Número de componentes de código
- Número de pruebas unitarias
- Número de pruebas de integración
- Número de pruebas de sistema
- Tamaño del código fuente (líneas efectivas de código + líneas de comentarios + líneas en blanco)
- Tamaño de la documentación producida (número de páginas de cada uno de los documentos entregables)

→ Tamaño del código

$$LOC = \sum \text{sentencias_ejecutables} + \sum \text{sentencias_comentadas} + \sum \text{líneas_en_blanco}$$

→ Número de problemas y discrepancias

Se mide su tendencia a lo largo del ciclo de vida en cuanto a:

Problemas/Discrepancias (abiertos, cerrados, descartados) tanto de código como de documentación

→ Estabilidad de los requisitos

$$\text{Estabilidad} = 1 - \frac{\sum \text{Reqs_modificados} + \sum \text{Reqs_borrados} + \sum \text{Reqs_nuevos}}{\sum \text{Requisitos}}$$

→ Número de sentencias por función

$$N_STMITS = \sum \text{sentencias_ejecutables}$$

El umbral establecido para este valor fue de 150 líneas

→ Número de sentencias por módulo

$$N_STMITS = \sum \text{sentencias_ejecutables}$$

El umbral establecido para este valor fue de 1.500 líneas

→ Complejidad ciclomática

$$V(G) = \text{Número ciclomático de McCabe} = A - N + 2$$

Donde:

- A: Arcos del grafo de flujo de la función, y
- N: Nodos del grafo de flujo de la función

El valor máximo establecido por McCabe para este indicador es de 10. Por encima de este valor y como se ha explicado anteriormente, resulta extremadamente difícil probar la función

→ Frecuencia de comentarios

$$COM_R = \frac{N_COM}{N_STMITS}$$

Donde:

- N_COM es el número de líneas de comentarios en el código, y
- N_STMITS es el número de sentencias ejecutables

El umbral establecido para este indicador fue del 20% (0,2)

→ Número de niveles de anidamiento

MAX_LVL = número máximo de estructuras de control anidadas en una función

El umbral establecido para este indicador fue de 8

→ Número de puntos de salida

N_OUT = número de puntos de salida de una función

El umbral establecido para este indicador fue de 1

→ Progreso de pruebas unitarias

Para declarar que una prueba unitaria ha resultado exitosa es necesario que se haya cubierto el 100% de las sentencias de código, es decir, que todas las sentencias del código de la función han sido ejercitadas

$$UT_Progress = \sum \text{componentes_probados_unitariamente_con_éxito}$$

→ Progreso de pruebas de sistema

$$ST_Progress = \frac{\sum \text{pruebas_de_sistema_ejecutadas_con_éxito}}{\sum \text{pruebas_de_sistema_planeadas}}$$

→ Seguimiento de hitos

Seguimiento de hitos = fechas de hitos planificados - fechas de hitos actuales

→ Seguimiento de acciones

Tendencia del número de acciones (totales, abiertas y cerradas)

→ Seguimiento de riesgos

Tendencia del número de riesgos (totales, abiertos y cerrados)

Medición Práctica de Software y Sistemas (PSM)

Información objetiva para la toma de decisiones



Autor: Antonio Moya / Comité de Software de la AEC. Director Programa Excelencia en la Operación. Ericsson R&D Madrid

Publicado en la Revista CALIDAD Nº 6, SEPTIEMBRE 2006

Medición Práctica de Software y Sistemas (Practical Software & Systems Measurement) es un marco de referencia para la gestión por datos, especialmente indicado para jefes de proyecto software, basado en la integración de los conceptos de un Modelo de Proceso de Medición y de un Modelo de Información de Medida. PSM está patrocinado por el Departamento de Defensa de los EEUU y ha servido de base para la norma ISO/IEC 15939.

El enfoque de PSM se centra principalmente en la medición de los proyectos de software, donde las medidas ayudaran a los jefes de proyecto a realizar un trabajo más eficaz. Un buen sistema de medición permitirá una planificación más realista, una mejor asignación de los recursos necesarios, siempre limitados, para la realización de los planes, así como un mejor y más preciso seguimiento del progreso en la ejecución de dichos planes.

¿Por qué medir el software?

Hoy en día, el software se ha convertido en uno de los factores más importantes, si no el que más, en las inversiones de las compañías, así como en las estrategias de negocio. Esto es así debido a la importancia del software en la capacidad de una organización para poder mantener el ritmo en un entorno cada vez más competitivo y cambiante desde el punto de vista tecnológico. Por todo ello, cada día hay mayor necesidad, en las organizaciones, de disponer de un sistema de medición que les permita evaluar y gestionar de forma más objetiva los proyectos intensivos de software.

Un sistema de medición del software bien implantado permite relacionar e integrar la información derivada de proyectos anteriores de la organización y de las disciplinas de gestión técnica, proporcionando la información que permitirá a los jefes de proyecto tomar decisiones clave y definir acciones apropiadas para llevar a cabo dichas decisiones. En definitiva, lo que PSM proporciona a los jefes de proyecto, y a los gestores de la organización, es una información objetiva como base de las decisiones a tomar.

Por otro lado, un buen sistema de medición del software debe cumplir los requisitos siguientes:

- Proporcionar una comunicación efectiva reduciendo ambigüedades, y suministrando la información necesaria para identificar, priorizar, monitorizar y comunicar los objetivos y problemas asociados al proyecto a todos los niveles de la organización, incluyendo la comunicación con los clientes.
- Permitir el seguimiento de los objetivos de proyecto describiendo de forma precisa el estado tanto de los productos como de los procesos implicados en el proyecto software. Debe proporcionar la información precisa para responder a preguntas tales como: ¿progresa el proyecto de acuerdo al plan?, ¿los costes están dentro del presupuesto planificado?, ¿está el producto preparado para su entrega al cliente?
- Identificar y corregir los problemas cuanto antes, facilitando la gestión proactiva, usando las medidas como recurso para evitar entrar en un enfoque reactivo de reparación de fallos, una vez producidos éstos. Las medidas permiten detectar los problemas antes y, por tanto, corregirlos cuando su impacto es menor tanto por la dificultad de su resolución, como de coste de reparación.
- Poder tomar decisiones de compromiso entre las diferentes restricciones a las que normalmente está sujeto un proyecto software: coste, tiempo, calidad capacidad, rendimiento, etc. Las medidas deben ayudar a determinar más objetivamente los impactos, en otras áreas, debidos a las decisiones tomadas para cumplir con las restricciones en un área.
- Apoyar la justificación de decisiones proporcionando a los jefes de proyecto y gestores de la organización datos históricos de rendimiento sobre los que basar sus estimaciones y planes, así como los fundamentos para elegir las mejores alternativas en caso de necesitar cambios en la planificación.

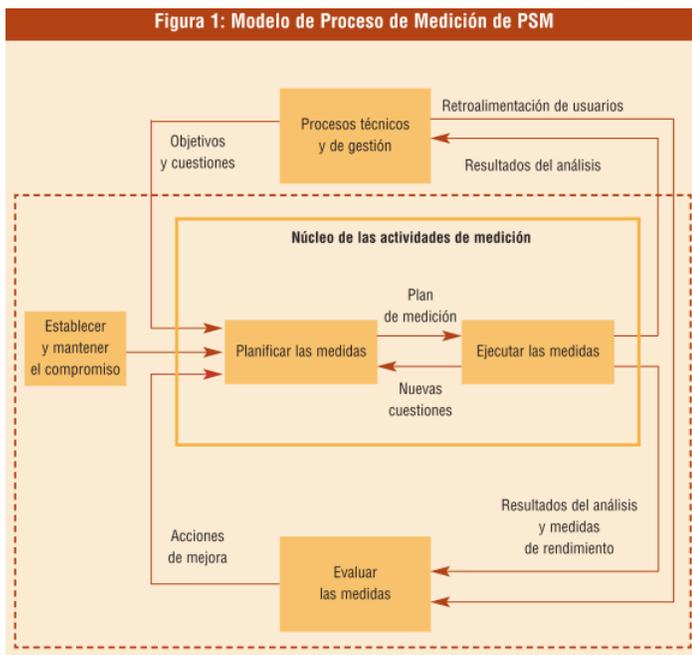
PSM satisface todos estos requisitos anteriores proporcionando guías basadas en la experiencia de profesionales de medición del software, de cómo definir e implantar un proceso de medición dirigido por un enfoque en las necesidades de información de un proyecto intensivo de software.

PSM estructura la información de medidas de un proyecto software por medio de un Modelo de Información de Medidas, al tiempo que describe las actividades de medición usando un Modelo de Proceso de Medición.

Modelo de Proceso de Medición

Un buen modelo del Proceso de Medición debe contener todas las fases que se detallan en la figura 1.

La fase más importante, y la más difícil de implementar de todas, es la de Establecer y mantener el compromiso, puesto que implica conseguir el apoyo, tanto de la organización como del proyecto, a la realización de las restantes fases y la asignación de la infraestructura necesaria para la viabilidad y el éxito del Proceso de Medición. El núcleo de las



actividades de medición lo constituyen las fases de Planificación y ejecución de las medidas.

En la fase de Planificación se empezará por identificar las necesidades de información del proyecto y la selección de las medidas que harán posible tratar las cuestiones en aquellas áreas de interés de los jefes de proyecto. En esta fase se genera el Plan de Medición que recoge tanto la definición de las medidas, como de los procedimientos para la recogida de datos, su procesamiento y análisis, generación de informes y de evaluación del propio proceso de medición y de la idoneidad de las medidas en uso de la organización. En la de Ejecución se realiza la recogida de datos y su procesamiento, el análisis y la generación de los productos de información que darán satisfacción a las necesidades de información de los jefes de proyecto y de la organización.

En la fase de Evaluación de las medidas se hará el análisis de la capacidad del propio proceso de medición y de la idoneidad de las medidas actuales de la organización para la satisfacción de las necesidades de información. Si esta fase se realiza continuamente, ayudará a incrementar la madurez del Proceso de Medición de la organización. Las actividades de la fase de Procesos técnicos y de gestión no son de medición sino de interfaz con el proceso de medición para determinar qué necesidades de información hay para tomar decisiones, operando dentro de los procesos de gestión y técnicos. Sin embargo, por muy bueno que sea el sistema de medición, éste no puede garantizar el éxito de un proyecto, pero lo que sí permitirá a los jefes de proyecto es tener un enfoque proactivo en la gestión de los temas críticos de los proyectos de software.



Modelo de Información de Medidas

El Modelo de Información de Medidas es uno de los conceptos fundamentales de medidas de PSM, pues proporciona un mecanismo formal para ligar las necesidades de información con los procesos de ingeniería y los productos (entidades) que pueden ser medidos.

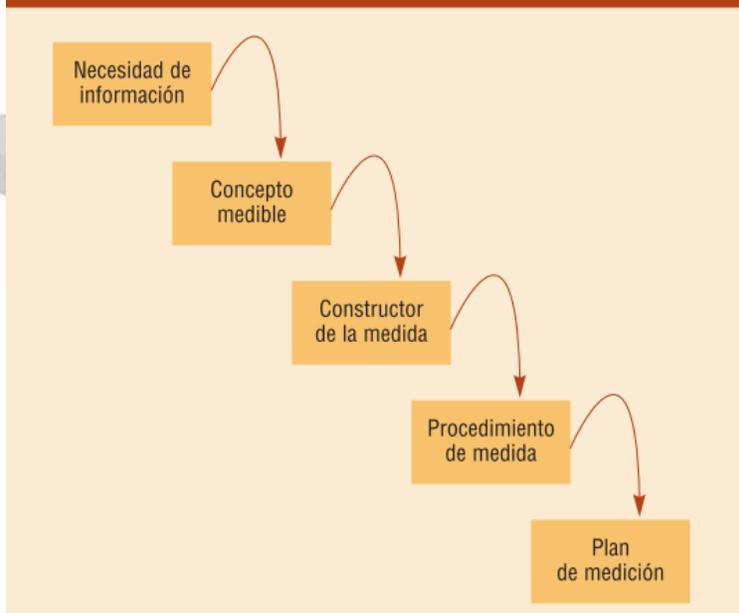
El modelo de información establece una estructura definida para relacionar diferentes conceptos y términos de medidas. En la figura 3 se muestra cómo a partir de una necesidad de información derivamos un plan de medición que lo satisfaga:

Todo plan de medición debe empezar a partir del reconocimiento de la necesidad de información requerida para la toma de decisiones en el proyecto. Los datos necesarios para satisfacer estas necesidades se obtendrán midiendo diferentes elementos software y las características del proceso que llamaremos entidades software.

El concepto medible es una idea acerca de qué entidad o entidades software serán medidas para la obtención de datos que satisfagan esa necesidad de información. Como ejemplo, el presupuesto y los recursos a asignar a un proyecto se pueden determinar mediante la productividad. Determinar la productividad requerirá identificar qué entidades deben medirse. Aquello que será medido y cómo deben combinarse los datos para producir los resultados que satisfagan la necesidad de información se materializará en un constructor de la medida. Para medir la productividad, en nuestro ejemplo anterior, el esfuerzo y el tamaño del

software podrían ser dos medidas a realizar. La forma en que se recogerán los datos y se organizarán se definen en un procedimiento de medición que instanciará el correspondiente constructor de medida. Finalmente, todas las necesidades, de información, conceptos medibles, constructores de medida y procedimiento de medición se recogerán en un plan de medición

Figura 3: Evolución de una necesidad de información en un plan de medición



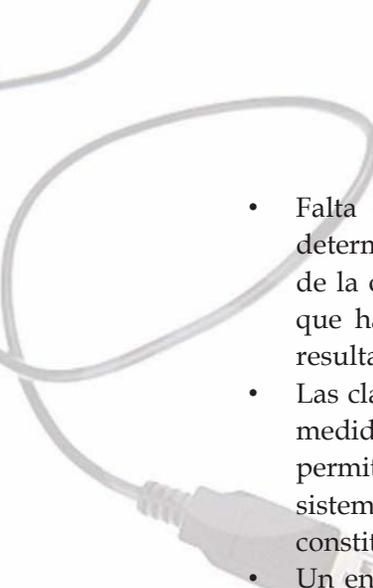
La ejecución del plan de medición producirá los productos de información que responden a las necesidades de información para la toma de decisiones.

Lecciones aprendidas

A través de la consolidación de prácticas y de experiencias a lo largo de años en muchas organizaciones intensivas de software, se han podido detectar muchos errores y también buenas prácticas en el campo de la medición del software, las cuales expongo a continuación.

Los errores más comunes son:

- Recoger muchísimos datos, que luego son raramente usados contribuyendo a reducir la moral de la organización y a un gasto enorme. Esta lección lleva a otra y es que las medidas cambian continuamente de unos años a otros, dada la insatisfacción de la organización, por encontrar aquellas medidas que le aseguren que están apuntando al objetivo adecuado.
- Incoherencia, ambigüedad e inconsistencia de las medidas junto con la falta de calidad y fiabilidad de los datos, lo que lleva siempre a cuestionar la validez de los análisis.
- Uso erróneo de las medidas para evaluar a las personas en lugar del rendimiento de la organización y de sus procesos. En este caso se producen los siempre válidos dichos de que “todo lo que se mide mejora” y “dile al manager lo que quiere oír”.
- Elegir las medidas erróneamente llevando a la organización a mejorar aspectos irrelevantes para los objetivos de negocio. En este caso se mide el rendimiento del proceso erróneo. Un ejemplo sencillo ocurriría si el objetivo de negocio fuera satisfacción de cliente a través de resolver los fallos del software lo más rápidamente posible, y elegimos como medida el tiempo medio que tardamos en resolver todos los fallos reportados. El resultado sería que los problemas sencillos, que suelen ser los más numerosos, se resolverían siempre antes.
- Producción inusitada e inconsistente de gráficos e indicadores, pero falta de análisis rigurosos, debido principalmente a la interpretación particular y no establecida de para qué se mide, cómo se debe interpretar cada medida, qué objetivo tiene la medida. Todo ello con el añadido de la falta de uso de datos históricos para caracterizar el comportamiento de los procesos de la organización y su utilización para detectar riesgos y tomar decisiones.

- 
- Falta de un experto en medidas del software, con conocimientos estadísticos, que determine la idoneidad de las medidas según los objetivos y necesidades de información de la organización, asegurando el correcto uso y el análisis adecuado de las medidas y que haga propuestas de actuación para la organización derivadas del análisis de los resultados de las medidas.
 - Las claves de éxito de un sistema de medición no residen en elegir el mejor conjunto de medidas para los proyectos de una organización, ni en el uso de herramientas que permitan la recogida y análisis de los datos automáticamente. Las claves de éxito de un sistema de medición residen, principalmente, en las características siguientes, que constituyen un conjunto de buenas prácticas avaladas por unos resultados excelentes:
 - Un enfoque dirigido por las necesidades de información de los responsables de tomar decisiones en los proyectos. Las medidas se definen y se implementan para tratar de proporcionar la información necesaria a los jefes de proyecto con el fin de resolver las cuestiones o problemas de los proyectos y alcanzar los objetivos establecidos. En este enfoque es fundamental entender las relaciones entre cuáles son las necesidades de información, lo que se está midiendo, y cómo estas medidas se traducen en resultados válidos para la toma de decisiones.
 - Un Proceso de Medición estructurado y repetible que defina las actividades de medición y las interfaces de información relacionadas. Este proceso debe ser flexible y adaptable a los procesos técnicos y de gestión usados en cada dominio de aplicación. Este proceso no debe ser un proceso estático sino iterativo, enfocándose siempre en las cuestiones más críticas, que van cambiando en la medida en que los objetivos del proyecto (en el que se implementa el Proceso de Medición) cambian y, por tanto, los problemas y necesidades de información.
 - Las medidas deben realizarse dentro un proyecto como un proceso de soporte a la ingeniería de software, lo que implica que el proyecto debe incluir todas las actividades de planificación y ejecución de las medidas, así como las actividades de análisis de los resultados y evaluación de las actividades de medición. El resultado de este proceso debe ser un conjunto de productos de información (datos, indicadores, análisis y acciones sobre desviaciones) que soporten las necesidades de información del proyecto u organización.
 - La organización debe empezar con un conjunto básico de medidas que debe ir ampliando poco a poco en el camino a la excelencia. Este conjunto básico se ampliará en la medida en que la organización tenga unas nuevas necesidades de información, y aplicando continuamente la fase de Evaluación de las medidas.
 - En el camino a la excelencia, lo que realmente da valor no es el hecho puntual de la medida, sino las tendencias que muestran los indicadores a lo largo del tiempo.

Estas tendencias son las que nos marcarán si la organización progresa hacia los objetivos o si hay desviaciones que se deben analizar en profundidad.

PSM satisface todas estas características proporcionando guías basadas en la experiencia de profesionales de medición del software, de cómo definir e implantar un proceso de medición, definiendo medidas y sus constructores, proporcionando métodos de análisis, y todo ello soportado por una herramienta gratuita que facilita la implantación del proceso de medición y al gestión de las medidas en los proyectos y en la organización. Para más información visitar la página web de PSM: www.psmc.com



personas



procesos



tecnologías

Testing

Parece mentira que en cualquier ámbito de la sociedad industrial las pruebas son un proceso elemental y respetado en cualquiera de sus ámbitos, sin embargo el sector de las Tecnologías de la Información, en su aspecto de lo que podíamos denominar Informática de Gestión tiene un peculiar “Talón de Aquiles”. Aquí se detallan tres artículos que manifiestan esas situaciones y algunas soluciones que miembros del comité detallan con experiencia en sus organizaciones.

El propósito de las pruebas es reducir el riesgo inherente en los sistemas, mediante la identificación temprana de problemas.

El objetivo de identificar problemas no es otro que arreglarlos, obteniendo como resultado un producto fiable y libre de fallos. Aunque cuando se prueba un programa se tiene una actitud destructiva hacia éste, ya que se trata de mostrar sus defectos, en el contexto global del proyecto el carácter es plenamente constructivo, ya que se trata de evitar que tales defectos sobrevivan hasta el producto final.

La actividad de pruebas se entiende como validación de un resultado a través de su accionamiento para verificar que responde conforme está prevista. En tal sentido no solo se puede ejecutar el código, para ver su funcionamiento, sino también otros productos del desarrollo tales como requisitos, especificaciones, diseños, pueden ser sometidos a prueba lógica para verificar que realizan todo aquello que cabe esperar de los mismos. De este modo se entienden las pruebas, en su sentido más amplio, como evaluación del producto a lo largo de su desarrollo.

Debido a su complejidad, no es posible probar, de forma económica, un programa completamente, por lo que es necesaria una gestión de pruebas efectiva que asegure el cumplimiento de los objetivos de las pruebas a un coste razonable.

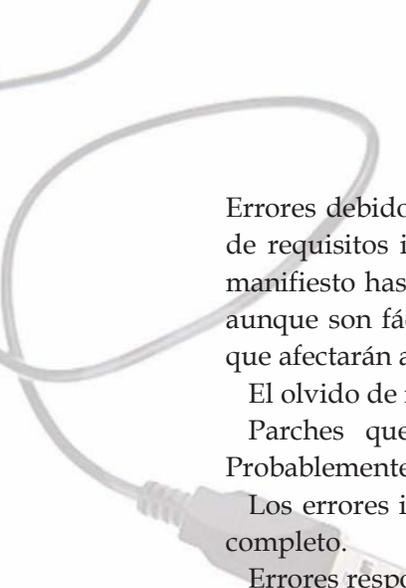
La misión principal de las pruebas en las organizaciones es desarrollar un nivel objetivo de confianza sobre la calidad de los productos, acorde con las expectativas de cada cliente.

La gestión de las pruebas persigue los siguientes objetivos:

- Detectar los defectos y neutralizar sus efectos lo antes posible.
- Realizar actividades de evaluación en todas las fases del Ciclo de Vida.
- Planificar las actividades de evaluación y pruebas.
- Registrar y controlar los resultados de las actividades de evaluación.
- Conocer en todo momento los defectos detectados pendientes de resolver.
- Establecer criterios objetivos de aceptación de los diferentes productos.

Estos objetivos adquieren una trascendencia fundamental en el caso de los desarrollos complejos en los que la identificación, localización y eliminación de cada defecto se complica conforme el desarrollo avanza y existe un mayor número de interdependencias entre sus componentes, dando lugar a problemas como:

Los errores iniciales se transmiten fase tras fase hasta que son detectados en el producto final.



Errores debidos a malas interpretaciones de la necesidad del cliente que motivan la inclusión de requisitos inadecuados que incorporados al diseño se implementan y no son puestos de manifiesto hasta que el cliente recibe el producto final. Son errores que una vez identificados aunque son fáciles de aislar darán lugar a un importante esfuerzo de desarrollo adicional ya que afectarán a muchos elementos del producto interrelacionados.

El olvido de requisitos que supone la incorporación de parches tardíos en el producto.

Parches que, además de suponer trabajo no previsto, serán difíciles de mantener. Probablemente serán fuente de errores adicionales inducidos.

Los errores introducidos en el diseño detallado de módulos que supone rehacer el módulo completo.

Errores responsables de un elevado porcentaje de desviación en la planificación del proyecto ya que supone repetir trabajo que no se realiza bien a la primera.

Los pequeños errores ocultos que tras las pruebas de integración provocan condiciones de fallo escasamente controlables.

Errores de muy difícil localización, que requieren un desmesurado esfuerzo hasta su correcto aislamiento, dando lugar a que frecuentemente permanezcan abiertos durante mucho tiempo.

Los errores que perduran en el producto final instalado en campo cuyo coste de corrección se multiplica.

Fallos que pueden requerir el desplazamiento a lugares remotos y condiciones difícilmente reproducibles originando un elevado coste de diagnóstico e identificación.

El objetivo final de la Gestión de Pruebas es evitar estos problemas, logrando que los defectos no se propaguen a través de las fases de desarrollo y alcancen al producto final.

Los modelos de prueba y evaluación de producto se aplican a los productos resultantes de todas las fases y actividades del ciclo de desarrollo de producto. Desde el punto de vista de las pruebas se consideran tres grandes categorías de productos que son tratadas de modo diferenciado en su evaluación:

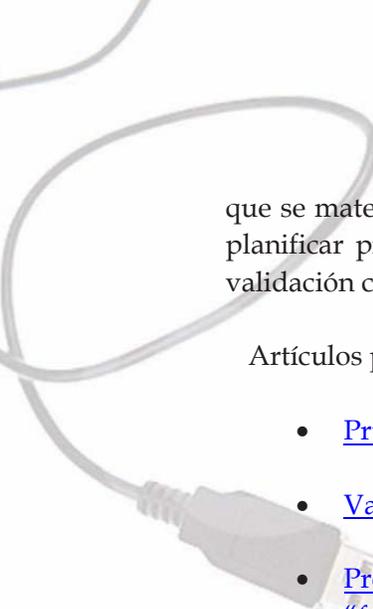
Categoría Uno: Software. Tiene la característica de poder ser ejecutado para experimentar con su funcionalidad. Éstos serán objeto de dos tipos de pruebas, en su versión “fuente”, serán sometidos a actividades de revisión e inspección, y en su versión “ejecutable” a actividades de prueba funcional.

Categoría Dos: Documentos y registros de información. Estos únicamente podrán ser objeto de actividades de tipo revisión e inspección. Pero en el caso de las revisiones de pruebas estas se orientaran como pruebas de simulación de casos, con formulación de tipo condicional del estilo: si de hace... / sucede..., o aplicando lo documentado sucedería.

Categoría Tres: Hardware, firmware, comunicaciones y otros elementos relacionados con Sistemas Operativos, Bases de Datos y plataformas en general. Deben ser comprobados, validados y certificados

Por otra parte, es de gran importancia que el desarrollo de las pruebas, desde su planificación hasta su ejecución, siga fielmente los pasos de la integración de producto. El objetivo de la integración es construir el producto final con una adecuada estructura modular orientada a facilitar su desarrollo y mantenimiento, tal que facilite la construcción progresiva, de modo acumulativo, de la funcionalidad del producto final.

Las actividades de pruebas deben estar presentes en cada uno de esos pasos de integración,



que se materializarán mediante entregas separadas para pruebas. El proceso de pruebas debe planificar pruebas concretas para cada una de estas entregas y, en su momento, someter a validación cada entrega.

Artículos presentados en esta selección:

- [Pruebas automatizadas y calidad para aplicaciones y sitios web](#)
- [Validación del producto software](#)
- [Prevención de Defectos: un modelo de análisis de productos potencialmente "fétidos" \(stinkers\)](#)
- [La especialización de las pruebas del software.](#)



personas



procesos



tecnologías

Pruebas automatizadas y calidad para aplicaciones y sitios web



Pruebas automatizadas y calidad

La audiencia instantánea de una aplicación alojada en un sitio web, o del sitio web en sí mismo, hace que su calidad y fiabilidad sean los factores cruciales para su éxito. La naturaleza de los sitios y las aplicaciones web, presenta grandes retos en el ámbito de las pruebas del software. Los webmasters, desarrolladores de aplicaciones web y los gestores del control de calidad de los sitios web, necesitan herramientas y métodos que satisfagan sus necesidades específicas. Las pruebas automatizadas específicas para entornos web, basadas en navegadores web, ofrecen una solución clara para garantizar la calidad web.



Introducción
Los entornos web suponen un reto absoluto en el mundo de la calidad del software. A los pocos minutos de vida, una aplicación web puede ser accedida por miles de usuarios más que si fuese una aplicación no web. La inmediatez del web, crea una necesidad inmediata de calidad y entrega rápida de la aplicación, pero la complejidad técnica del web y las variaciones entre navegadores hacen que las pruebas y el control de calidad sean mucho más difíciles que las aplicaciones convencionales o las "cliente / servidor". Existe un reto importante en las pruebas automatizadas de entornos web.

Definición de calidad y fiabilidad web
No existe una medida única de la calidad de un sitio o aplicación web, de la misma manera que en otro software complejo. Descubrimos las siguientes dimensiones de la calidad web, y las medidas que ayudan a caracterizarla:

• **Temporalidad:** los webs cambian rápido y a menudo. ¿Cuántos cambios se han

Autor: Ramiro Carballo Gutiérrez, Presidente del Comité de Software de la AEC y Director Comercial de Gesein

Publicado en la Revista Calidad - Julio-Agosto 2005

La audiencia instantánea de una aplicación alojada en un sitio web, o del sitio web en sí mismo, hace que su calidad y fiabilidad sean los factores cruciales para su éxito. La naturaleza de los sitios y las aplicaciones web, presenta grandes retos en el ámbito de las pruebas del software. Los webmasters, desarrolladores de aplicaciones web y los gestores del control de calidad de los sitios web, necesitan herramientas y métodos que

satisfagan sus necesidades específicas. Las pruebas automatizadas específicas para entornos web, basadas en navegadores web, ofrecen una solución clara para garantizar la calidad web.

Introducción

Los entornos web suponen un reto absoluto en el mundo de la calidad del software. A los pocos minutos de vida, una aplicación web puede ser accedida por miles de usuarios más que si fuese una aplicación no web. La inmediatez del web, crea una necesidad inmediata de calidad y entrega rápida de la aplicación, pero la complejidad técnica del web y las variaciones entre navegadores hacen que las pruebas y el control de calidad sean mucho más difíciles que las aplicaciones convencionales o las "cliente / servidor". Existe un reto importante en las pruebas automatizadas de entornos web.

Definición de calidad y fiabilidad web

No existe una medida única de la calidad de un sitio o aplicación web, de la misma manera que en otro software complejo. Descubrimos las siguientes dimensiones de la calidad web, y las medidas que ayudan a caracterizarla:

- **Temporalidad:** los webs cambian rápido y a menudo. ¿Cuántos cambios se han producido desde la última actualización? ¿Cómo se resaltan las partes que han cambiado?
- **Calidad estructural:** ¿qué tal se integran todas las partes del web? ¿Funcionan todos los enlaces internos y externos? ¿Se ven todas las imágenes? ¿Existen partes inconexas en el web?

- Contenido: ¿se ajusta el contenido de páginas críticas a lo que se supone que debían contener? ¿Mantienen estas páginas la calidad de contenidos en los cambios de versión? ¿Funcionan correctamente las páginas dinámicas?
- Precisión y consistencia: ¿son suficientemente precisos los datos ofrecidos al usuario? ¿Son consistentes con los presentados en otras ocasiones? ¿Cómo lo sabemos?
- Tiempos de respuesta y retardos: ¿Responde el servidor web a las peticiones del cliente web dentro de ciertos parámetros de rendimiento? En un contexto de comercio electrónico, ¿cómo es el tiempo de respuesta extremo-a-extremo después de un SUBMIT? ¿Existen partes del web que son tan lentas que el usuario las abandona?

El impacto de la calidad es muy importante. La calidad reside en la mente del usuario. Un sitio o aplicación web de baja calidad, con muchos enlaces rotos, con falta de algunas imágenes, con mensajes de error tras la ejecución de ciertos CGI-BIN's, etc., puede suponer un gran coste si se cuantifica el empobrecimiento de las relaciones con el cliente, pérdida de imagen corporativa, e incluso en pérdidas en los ingresos por ventas. Los sitios web muy complejos y desorganizados, pueden desbordar a los usuarios.

La combinación de la complejidad del web y la baja calidad, es mortal para los objetivos de una compañía. Los usuarios insatisfechos se decantarán rápidamente por un web diferente, y seguramente se marchen con una mala impresión.

Factores de la arquitectura web

La complejidad del web, a pesar de que le proporciona potencia, puede también impedir que se asegure su calidad. Si añadimos páginas compuestas por varios sitios web, las actualizaciones rápidas, etc., el problema se incrementa.

Para solucionarlo, debemos analizar cada componente del web, garantizar que no supondrán un problema.

Navegador

Uno de los principales componentes de un sitio o aplicación web, desde el punto de vista de la calidad es el navegador. Existen muchos distintos, o con distintas configuraciones, en los que un sitio bien construido debería verse bien. Para conseguir esto, el web debe usar sólo aquellas construcciones que permitan visualizarlo en la mayoría de los navegadores. Sin embargo, esto está reñido con la creatividad, y pocas veces se cumple, excepto si se asume una elevada complejidad técnica y se gestiona que se debe comprobar en cada uno de los tipos de navegadores.

Tecnologías de visualización

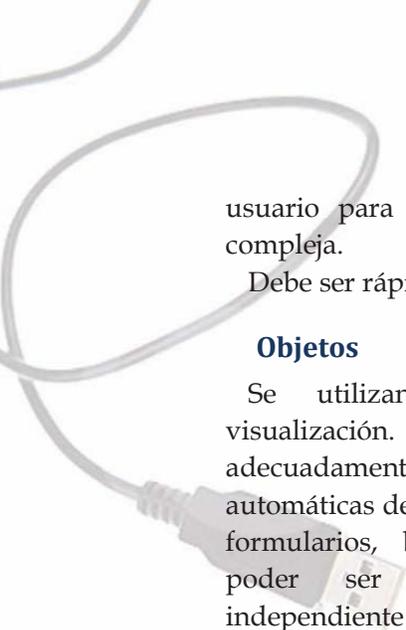
Otro componente son las tecnologías de visualización, que obligan a comprobar que

el sitio es compatible con ellas:

- HTML, existen varias versiones.
- Java, JavaScript, ActiveX: hay que asegurar el funcionamiento de estos elementos de ejecución en el cliente, principalmente los ActiveX, son bastante problemáticos.
- Scripts Cgi-Bin, incluyendo perl, awk, shell-scripts, etc., como elementos de ejecución en el servidor. Se debe garantizar que son correctas las operaciones de extremo a extremo.
- Accesos a bases de datos. Donde la complejidad se corresponde a la de una aplicación transaccional pura, debiéndose garantizar que se obtienen las respuestas esperadas para las entradas de datos correspondientes.

Navegación

La secuencia de clicks que realiza un



usuario para recorrer el web, suele ser compleja.

Debe ser rápida y libre de errores.

Objetos

Se utilizan para configurar una visualización. Deben ser manejados adecuadamente por las herramientas automáticas de pruebas web, como enlaces, formularios, botones, tablas, etc., para poder ser validados de manera independiente a su representación.

Respuesta del servidor

De la velocidad del servidor depende la rapidez con la que el web se visualice. Evidentemente, influyen otros factores, como la carga de Internet, pero no tenemos solución para este tipo de problemas.

Requisitos para la automatización de pruebas web

Garantizar la calidad del sitio y aplicaciones web requiere la ejecución de conjuntos de pruebas, automática y repetiblemente, que demuestren las propiedades y comportamientos requeridos. Algunos elementos que deben tener las herramientas de pruebas para conseguir esto son:

Sesiones de pruebas

Las herramientas de pruebas web deben tener estas características:

- Independencia del navegador. Si dependemos de un navegador concreto, podemos estar enmascarando problemas del web.
- Sin caché ni búferes. El caché local y el búfer, utilizados para mejorar la sensación de rendimiento, se debería deshabilitar para conseguir que los tiempos medidos correspondan exactamente a la interacción navegador-sitio web-navegador.
- Fuentes y preferencias. No deben afectar al modo en que se asegura la calidad.
- Objetos. Editar campos, pulsar botones, etc., debe poder ser gestionado sin problemas.

Sin embargo, si se trata de un sitio muy popular, o una aplicación muy utilizada, es importante ajustar el servidor mediante la realización de pruebas de carga.

Interacción y realimentación

Si se trata de un sitio pasivo, que sólo alberga contenidos, es suficiente que estén constantemente disponibles.

Si se trata de un sitio que interactúa con el usuario, el factor de calidad más importante es la rapidez y fiabilidad de esta interacción.

Usuarios concurrentes

Una inadecuada gestión de la concurrencia originada por navegadores situados en distintas localizaciones genera mucha complejidad en las pruebas.

La capacidad de la herramienta para operar en modo Objetos, sin que le afecten las fuentes o las preferencias, es esencial para garantizar que las pruebas se realizarán igual aunque los objetos cambien de posición en la pantalla. Pero la herramienta detectará como un defecto la ausencia de un botón que fue incorrectamente eliminado.

- Tablas y formularios. Las pruebas se deben ejecutar de la misma manera, aunque las tablas y los formularios se visualicen de manera distinta en distintos navegadores.
- Frames. No debe afectar a las pruebas el número de frames de una página.

Contexto de pruebas

Las pruebas se deben ejecutar desde el nivel de navegador, porque:

- Desde este nivel, los usuarios

visualizan el sitio o la aplicación. Por lo tanto, es la manera más realista de probar el web.

- Las pruebas ejecutadas desde el nivel de navegador se pueden ejecutar tanto webs locales como contra el web real. La

Validación dinámica del web

La clave de la calidad del web es asegurar la validez de lo que se ha probado. Éstas son 4 áreas donde la automatización de pruebas tiene un impacto significativo:

Pruebas operacionales

La fase de pruebas individuales o unitarias debe incluir una serie de comprobaciones sobre páginas individuales:

- Consistencia de la página. En qué se diferencia ésta de su versión anterior.
- Consistencia de tablas y formularios. Se garantiza que estén completas, que se presenten correctamente, etc.
- Relaciones entre páginas. Se pretende localizar enlaces rotos, enlaces que han desaparecido o se han añadido respecto de la versión anterior, etc.
- Tiempos de respuesta y consistencia del rendimiento. Se detecta la variación en rendimiento entre distintas versiones del web.

Suites de pruebas

Se permiten ejecutar pruebas en una variedad de modos:

- Pruebas desatendidas. Ejecutadas desde estaciones de trabajo.
- Pruebas en segundo plano. Se deben poder ejecutar pruebas simultáneamente en distintos navegadores que corran en background.
- Pruebas distribuidas. Para facilitar las pruebas de partes independientes del web.
- Pruebas de rendimiento. Se obtienen importantes datos de tiempos de respuesta medios.

ejecución local es perfecta para el control de la calidad, pero no para medir el rendimiento, donde son esenciales las medidas del tiempo de respuesta incluyendo retardos variables de Internet e interferencias del mundo real.

- Pruebas aleatorias. Debe existir esta posibilidad de pruebas por muestreo.
- Recuperación ante errores. No es normal que unas pruebas que simulan la acción de un usuario provoquen un fallo en el navegador, pero, si fuese así, la herramienta debería restablecer el sistema.

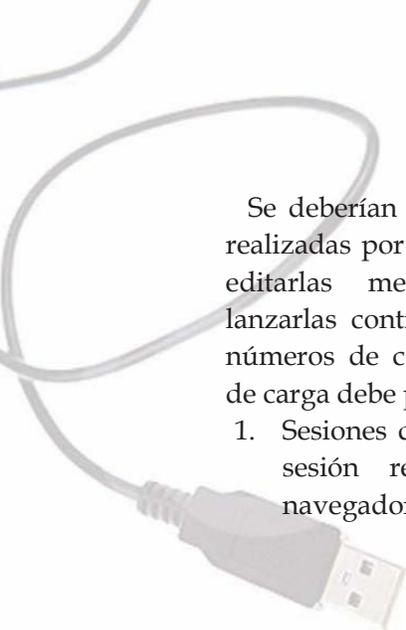
Validación de contenidos

Independientemente de cómo responda un web, el contenido debería ser comprobable, exacta o aproximadamente. Esto se puede garantizar de los siguientes modos:

- Estructural: se busca que los enlaces y anchos se ajusten a la línea base de datos anterior. Cada imagen se puede identificar mediante la medida del número de bytes, el tipo de fichero y otras propiedades.
- Puntos de comprobación y reproducción exacta. Uno o varios elementos del texto se pueden marcar como punto de comprobación para reproducción exacta.
- Estadísticas principales (línea, palabra, número de bytes, checksum, etc.).

Simulación de carga

El análisis de carga debe reproducir el comportamiento humano para ser realista, en lugar de enviar chorros de peticiones al servidor, simulando situaciones que nunca se van a dar en la realidad.



Se deberían grabar las sesiones en vivo, realizadas por un usuario, e incluso poder editarlas mediante scripts, y luego lanzarlas contra el servidor en diferentes números de concurrencias. La generación de carga debe proceder de:

1. Sesiones de un único navegador. Una sesión reproducida en un único navegador, con una o múltiples

respuestas. Los datos sobre tiempos, se guardarían en un fichero separado para su análisis.

2. Sesiones de múltiples navegadores independientes. Análogo a la anterior, permite aplicar métodos de análisis de multivarianza estadística para obtener un modelo de rendimiento completo.



personas



procesos



tecnologías

Validación del producto software



Autor: Luisa Morales Gerente de la Línea V&V. División de Consultoría de PROFIT. Miembro del Comité de Software de la AEC

Publicado en la Revista Calidad – Septiembre 2005

Problemática actual

Uno de los principales obstáculos a los que se enfrentan los proyectos de software es su finalización. En primer lugar, porque deben cumplir una fecha que siempre se ve demasiado ajustada, evitar sobrepasar las estimaciones de esfuerzo y coste y, sobre todo, porque el producto final debe ser aceptado por el grupo que concibió los requisitos para el mismo. Finalmente, la puesta en producción viene forzada por una necesidad de

contar con el producto funcionando, con independencia del nivel de calidad obtenido.

Varios son los factores que influyen en la buena conducción de los proyectos y en la consecución de productos de calidad, extendiendo la calidad no sólo a los aspectos técnicos sino también al fin último del propio producto, aquél para el que fue concebido.

En este artículo vamos a centrarnos en la validación del producto final, aspecto descuidado en muchos casos porque la responsabilidad de validar y, por tanto, de aceptar el producto software es del cliente interno o externo para el que se ha abordado el proyecto. Esta división de responsabilidades conlleva errores de concepto, que es imprescindible conocer y solventar, si queremos obtener un éxito real a la finalización del proyecto.

- Primer error: "Primero yo construyo y después tú validas".
- Segundo error: Asumir que el usuario, además de conocer su negocio, domina las técnicas de validación.

En muchos casos, los proyectos son fuertes en prácticas como la aplicación de una metodología de desarrollo, definición del ciclo de vida de desarrollo, planificación detallada, asignación de roles, etc., pero son menos fuertes en la aplicación de una metodología específica de pruebas, no se define el ciclo de vida de las pruebas, y se dan importantes carencias en la planificación de las mismas, asignación de roles, etc. Como consecuencia, las pruebas se han tratado como una fase más dentro del ciclo de vida de desarrollo, a la que se llega, en muchas ocasiones, tarde, fuera de presupuesto, y con fuertes presiones para su finalización.

Esta fase de pruebas ha sido una fase a menudo descuidada y, casi siempre, sacrificada ante las presiones sobre el plazo o coste de los proyectos, su realización práctica ha sido desigual y ha estado a menudo supeditada a las presiones de plazo y de costes provenientes de otras fases

del ciclo.

Frecuentemente se ha incurrido en carencias de planificación de las pruebas, se ha reducido el alcance de éstas a un mínimo, fijado más por razones de plazo que por la importancia de la aplicación, y no se ha realizado una documentación adecuada de todo el proceso de pruebas. Como consecuencia, muchas pruebas han tenido lugar de facto como parte de la implantación y del rodaje, manifestándose, como fallos, los defectos existentes en el software, y cuya resolución en estas fases tardías ha producido un incremento sustancial del plazo y del esfuerzo respecto de los inicialmente previstos para el proyecto, y, en muchos casos, un efecto negativo sobre la imagen y confianza respecto de la función informática.

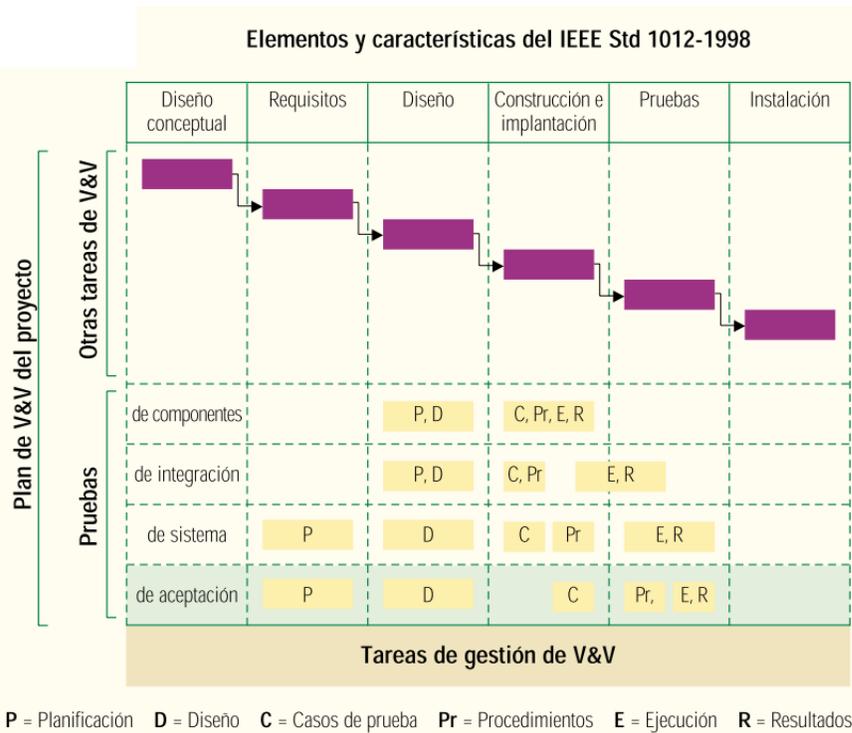
En este contexto, evidentemente, resulta muy difícil, sino imposible, conseguir una aceptación formal por parte del usuario antes de la puesta en producción del sistema.

Un enfoque diferente para la validación del producto software

La validación del software debe basarse en la prevención y detección temprana de los errores, ya que éstos se producen más frecuentemente en las primeras fases del ciclo de vida y el coste de corregirlos crece exponencialmente según avanza el proyecto. Este enfoque supera el concepto tradicional de pruebas, englobándolo en otro más amplio de Verificación y Validación (V&V), extendido a todas las fases del ciclo del desarrollo. Está encuadrado en un modelo de referencia de la función de V&V, basado en el estándar IEEE Std. 1012-1998 que ha sido desarrollado por el Comité de Estándares de Ingeniería de Software, perteneciente al Institute of Electrical and Electronics Engineers de Nueva York. Contempla todas las fases del proceso de desarrollo y considera también otros procesos software (adquisición, suministro, operación...), manteniendo en todo momento la perspectiva del negocio de la organización.

Este modelo cumple los requisitos que el Software Engineering Institute establece en su modelo CMMI para verificación y validación; incorpora una visión de sistemas (incluye la evaluación de aspectos tales como riesgos, fuentes de peligro, migración, retirada); cumple con otros estándares IEEE y con la norma ISO/IEC

Std 12207, relativos al ciclo de vida del software; aplica a todos los tipos de procesos SW; define cuatro niveles de integridad de las aplicaciones; establece un conjunto mínimo de tareas de V&V para cada una de las actividades del ciclo de vida de cada proceso SW; y proporciona criterios para cada tarea de V&V:



- a) corrección
- b) exactitud

- c) coherencia
- d) integridad

- e) legibilidad
- f) cualidad de poder ser probado

El éxito de la validación de los requerimientos reside en unas pruebas de aceptación completas e independientes.

El proceso de pruebas debe iniciarse en una fase temprana del ciclo de vida, integrándose en el desarrollo de la aplicación. Ambos procesos, desarrollo y pruebas, se realizan en paralelo, manteniendo un compromiso y una comunicación fluida entre los equipos de desarrollo y de pruebas. Las principales características que contribuyen a asegurar una validación exitosa de los requerimientos son:

- Las pruebas no son una fase aislada del proyecto, sino que constituyen en sí un proyecto con su propio ciclo de vida integrado en el del propio proyecto.
- Los equipos de desarrollo y pruebas son independientes, con funciones y perfiles diferenciados.
- Antes de la ejecución de las pruebas, se lleva a cabo todo un proceso metodológico que facilita y asegura el éxito de las mismas. En el momento de la ejecución, está todo previsto, tanto los aspectos funcionales como técnicos, evitándose la improvisación.
- La figura del usuario (equipo) participa en el inicio y fin del ciclo. La clave está en determinar los roles que participan: negocio, áreas técnicas (informática y organización), sistemas y producción.
- Se prueban todos los tipos de requerimientos del proyecto: funcionales, técnicos, de interfaz de usuario, de rendimiento, etc.

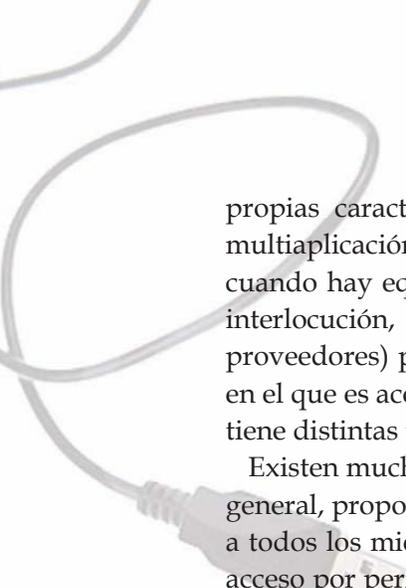
Equipo de pruebas

Las pruebas deben realizarse por un equipo independiente y multidisciplinar, con la siguiente composición:

- a) Un coordinador de pruebas para interactuar con los diferentes equipos involucrados:
- b) desarrollo, pruebas y usuarios funcionales y técnicos.
- c) Equipo de pruebas para realizar todas las actividades relacionadas con las pruebas de aceptación.
- d) Usuarios funcionales, responsables de cada parte funcional de las aplicaciones, y usuarios técnicos, responsables de los aspectos técnicos de las aplicaciones y la instalación, con las siguientes funciones:
 - a. Participación en la definición del Plan de pruebas.
 - b. Participación en la definición del Modelo de pruebas.
 - c. Participación en la ejecución de las pruebas.
 - d. Colaboración en el análisis de las incidencias.
 - e. Participación de los equipos de desarrollo objeto de las pruebas de aceptación.
 - e. Aceptación del producto final.
 - f. Un responsable de entornos de la instalación, con las siguientes funciones:
 - g. Participación en la planificación del entorno de pruebas.
 - h. Preparación del entorno de pruebas.
 - i. Soporte al entorno.
 - j. Limpieza del entorno de pruebas.

Herramientas para la gestión de pruebas

En muchos casos, la gestión de las pruebas de aceptación se soporta en una herramienta especializada. La decisión de utilizar una herramienta de gestión viene de la mano de las



propias características del proyecto. Se aconseja un apoyo de este tipo en los proyectos multiaplicación, que conllevan la involucración de distintos equipos de usuarios, también cuando hay equipos voluminosos de personas, que requieren mayor rigor en los procesos de interlocución, comunicación y cambios, o existen distintos equipos involucrados (varios proveedores) para los que puede haberse establecido acuerdos de nivel de servicio. Otro caso en el que es aconsejable el uso de una herramienta de gestión de pruebas es cuando el proyecto tiene distintas ubicaciones, con necesidad de compartir la misma información.

Existen muchas herramientas en el mercado orientadas a la gestión integrada de pruebas. En general, proporcionan un repositorio centralizado, se recomiendan que sean accesibles vía web a todos los miembros del equipo, desde cualquier punto geográfico, con diferentes niveles de acceso por perfil. Normalmente estas herramientas contienen los registros de los requisitos (en muchos casos, tienen facilidades de verificación de los mismos), ciclos, casos y pasos de pruebas asociados a los requisitos, e incidencias surgidas durante la ejecución de las pruebas. Estas herramientas aportan método al proceso de pruebas, mantienen la trazabilidad entre los requisitos, pruebas, resultados y anomalías, facilitan el trabajo de seguimiento de pruebas y la gestión de anomalías, y proporcionan todo tipo de estadísticas e información on line relevante para la gestión y seguimiento de las pruebas, evitándose la dependencia de envío de informes de situación, ya que la información está accesible para todos y los informes pueden ser solicitados por cualquier responsable. El grado de avance que resulta de la información que proporcionan es objetivo, ya que está basado en métricas.

Por último, el repositorio facilita información valiosa para posteriores pruebas de regresión y mantenimiento.

Las pruebas de aceptación contribuyen de forma decisiva a la consecución de los objetivos de calidad de los proyectos Como consecuencia de la adecuada validación de los requerimientos, los proyectos cumplen con su compromiso de calidad.

Los beneficios que se consiguen no sólo repercuten en el área inmediata de desarrollo de los proyectos, sino que se extienden al resto de la empresa, áreas de negocio y cliente final:

- “Informática” cumple sus compromisos de plazo, coste y calidad.
- Los proyectos pasan al entorno de producción con estabilidad, se minimiza el mantenimiento correctivo después de la implantación de los sistemas.
- Se ha dado respuesta a los aspectos técnicos relevantes del ciclo de vida en producción: crecimiento esperado de BBDD, puntos de backup y recovery, tiempos de respuesta, ventana batch...
- En definitiva, se facilita la transición de desarrollo a producción y se incrementa la confianza y el nivel de satisfacción de los usuarios finales de la explotación del nuevo sistema.
- Se incrementa la confianza y el nivel de satisfacción de los usuarios y se facilita el proceso de aceptación, ya que se proporciona una evidencia objetiva de la satisfacción de los requisitos del sistema.

Prevención de Defectos: un modelo de análisis de productos potencialmente “fétidos” (stinkers)

Prevención de Defectos: un modelo de análisis de productos potencialmente “fétidos” (stinkers)

El área de proceso de CMM “Prevención de Defectos” pertenece al nivel 5 de madurez porque es, en ese estadio de madurez de una organización, donde se obtiene la mayor eficacia, debido a que una organización, con ese nivel de madurez, siempre sabrá sobre qué parte de su proceso debe tomar medidas y qué impacto tendrán éstas, tanto en la calidad del producto como en el resultado del negocio, poniendo, por consiguiente, todos los medios suficientes y necesarios para que los defectos no vuelvan a aparecer. Pero eso no significa que no se deba abordar la prevención de defectos en etapas de madurez más tempranas, sobre todo si disponemos de una buena práctica avalada por resultados.

La prevención de la aparición de defectos es un círculo cerrado que se realimenta continuamente: detectado un defecto, analizo las causas de error que lo produce. Conocidas las causas de error, planifico acciones desde el inicio del diseño para que el defecto no se produzca de nuevo. Analizo el resultado de dichas acciones, determino su eficacia e identifico posibles nuevas acciones o detecto nuevas causas antes no identificadas. Para que este ciclo sea efectivo y eficiente, debe apoyarse en un modelo que soporte todas las fases del ciclo.

Este artículo propone un modelo de análisis relativamente sencillo que, se demuestra muy eficaz y que, además, permite ir continuamente mejorando la calidad de los productos.

Autor: Antonio Moya Catena Vicepresidente del Comité de Software de la AEC.

Publicado en la Revista CALIDAD N° 2, MARZO-ABRIL 2008

El área de proceso de CMM “Prevención de Defectos” pertenece al nivel 5 de madurez porque es, en ese estadio de madurez de una organización, donde se obtiene la mayor eficacia, debido a que una organización, con ese nivel de madurez, siempre sabrá sobre qué parte de su proceso debe tomar medidas y qué impacto tendrán éstas, tanto en la calidad del producto como en el resultado del negocio, poniendo, por consiguiente, todos los medios suficientes y

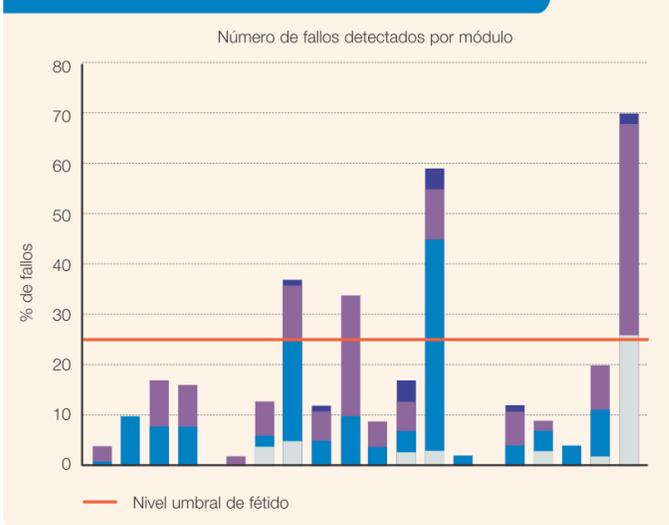
necesarios para que los defectos no vuelvan a aparecer. Pero eso no significa que no se deba abordar la prevención de defectos en etapas de madurez más tempranas, sobre todo si disponemos de una buena práctica avalada por resultados.

La prevención de la aparición de defectos es un círculo cerrado que se realimenta

continuamente: detectado un defecto, analizo las causas de error que lo produce. Conocidas las causas de error, planifico acciones desde el inicio del diseño para que el defecto no se produzca de nuevo. Analizo el resultado de dichas acciones, determino su eficacia e identifico posibles nuevas acciones o detecto nuevas causas antes no identificadas. Para que este ciclo sea efectivo y eficiente, debe apoyarse en un modelo que soporte todas las fases del ciclo.

Este artículo propone un modelo de análisis relativamente sencillo que se demuestra muy eficaz y que, además,

GRÁFICA 1. ESTADÍSTICAS DE NÚMERO DE DEFECTOS POR PRODUCTO



permite ir continuamente mejorando la calidad de los productos.

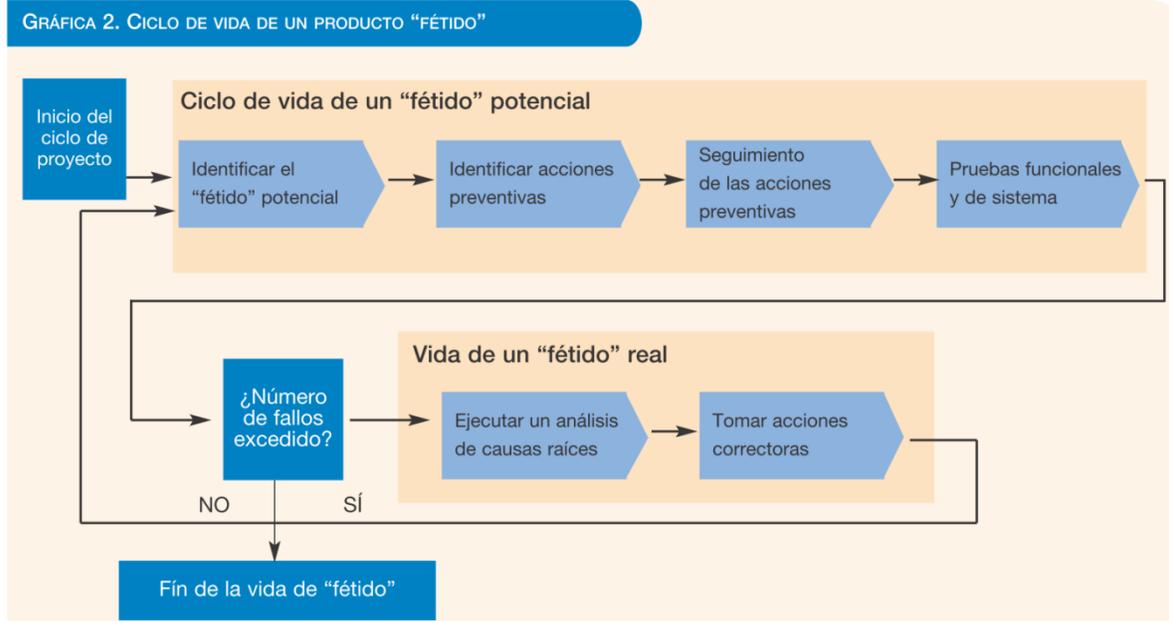
¿Qué es un producto “fétido”?

Producto “fétido” es todo aquel que supera un nivel de defectos (determinado por una organización) o que se prevé que, por diversos factores, pueda superarlo en caso de ser un producto nuevo o modificado. El nivel se puede establecer según distintos criterios: número de defectos del producto; número de fallos que producen; densidad de fallos; gravedad del fallo en términos de impacto en el servicio, por poner sólo algunos ejemplos.

En el caso práctico descrito en este artículo, el nivel lo establecemos por el número de defectos. La organización dispone de estadísticas sobre el número de defectos detectados en cada producto y, basándose en dichas estadísticas, establece el nivel actual. En la gráfica 1 se muestran estas estadísticas, así como el nivel a partir del cual un producto se denominará fétido.

En la gráfica 1 se observa que cuatro productos de los 20 representados superan el nivel de “fétido” (>25 defectos), por lo que en el próximo proyecto habrá que planificar las acciones oportunas para que deje de ser fétido.

Creación del modelo de análisis El modelo se construye a partir del análisis de las causas de error que producen los defectos (tanto en el código fuente, como en la documentación de operación y mantenimiento) y la identificación de los factores que intervienen en esas causas de error. A cada factor se le asigna una ponderación en función de su impacto y unos valores



para evaluar durante el análisis.

El modelo se construye siguiendo el siguiente ciclo de vida de un producto “fétido” (ver gráfica 2) en el que se distinguen dos caminos:

- Vida del producto posible “fétido” (o potencial fétido).
- Vida del producto que ya es “fétido”.

Durante el ciclo de vida de un producto ya “fétido” se obtienen los factores que intervienen en la producción de defectos. En este caso práctico se identificaron los siguientes factores agrupados en cuatro categorías:

- 1) Producto base
 - a) ¿Está liberado el producto base?

- b) ¿Cómo de crítico es el número de fallos o errores que aún quedan por reparar en el proyecto?
 - c) ¿Es el producto base ya un “fétido”?
 - d) ¿Cuál es el tamaño de la unidad software?
 - e) ¿Hay correcciones liberadas para el producto?
 - f) ¿Se liberó el producto como un diseño rápido?
 - g) ¿Cuál es la complejidad media del producto (McCabe por ejemplo)?
 - h) ¿El comportamiento del producto en campo depende de parámetros del sistema?
 - i) ¿Contiene el producto funcionalidades específicas de clientes o mercados?
- 2) Complejidad del diseño (Atributos genéricos)
- a) ¿Se va a diseñar el producto desde cero?
 - b) Si se modifica un producto existente, ¿cuál es el grado de modificación?
 - c) ¿Las estimaciones y el plan de tiempos son viables?
 - d) ¿Hay que modificar la documentación de cliente?
 - e) ¿Cuánta es la complejidad del impacto?
 - f) ¿Se van a modificar productos que interoperan con el producto bajo análisis?
 - g) ¿Cuándo y dónde se van a implementar los cambios anteriores?
 - h) ¿Son estables y claras las entradas para el diseño (no ambiguas)?
 - i) ¿Se han aprobado exenciones a los procesos?
 - j) ¿Se han aprobado peticiones de cambio para el producto?
 - k) ¿Contiene el producto más de una función?
- 3) Complejidad del diseño (Atributos dependientes de tecnología)
- a) ¿El producto funciona en tiempo real?
 - b) ¿Cuántos usuarios simultáneos deben servir?
 - c) ¿Interviene el producto en varias funciones principales?
 - d) ¿Maneja el producto comandos?
 - e) ¿Maneja el producto alarmas?
 - f) ¿Son complejos los comandos (en número de parámetros y opciones)?
 - g) ¿Gestiona ficheros el producto?
 - h) ¿Se puede quedar bloqueado el producto o los dispositivos que maneja?
 - i) ¿Interopera el producto con otros mediante señales?
 - j) ¿Se puede probar el producto completamente en un entorno simulado?
 - k) ¿Se probará el producto en un entorno real durante las pruebas de función?
 - l) ¿Se realizarán las pruebas básicas en un entorno simulado o real?
 - m) ¿Se tiene que implementar en el producto algún estándar?
- 4) Atributos de proyecto
- a) ¿Trabajarán más de una persona en la misma parte de diseño del producto?
 - b) ¿Codificarán más de una persona en la misma parte del producto?
 - c) ¿Estarán disponibles los autores de la base de diseño para consultas?
 - d) ¿Cuál es la experiencia media de los diseñadores con el producto?
 - e) ¿Estarán los autores del diseño de alto nivel disponibles para consultas de los diseñadores?
 - f) ¿Está la responsabilidad del diseño en otra organización o en el proyecto?
 - g) ¿Existen reglas de diseño y métodos de trabajo?
 - h) ¿Cómo es el plan de tiempos exigido al proyecto?
 - i) ¿Se aplicarán técnicas de trabajo en equipo?

- j) ¿Cuál es la experiencia del líder del equipo de diseño?
- k) ¿Tiene dicho líder habilidades de coordinación técnica?
- l) ¿El equipo de diseño es nuevo o ya conoce la funcionalidad?
- 5) ¿Está el proyecto impactado por el desarrollo de otras funcionalidades en otros proyectos?
- 6) ¿Se desarrollará parte del producto en otras organizaciones o suministradores?
- 7) ¿Formará parte el producto de varios sistemas, o de diferentes configuraciones?

Durante el ciclo de vida del “fétido” potencial se identifican acciones para evitar que el producto se convierta en “fétido” o siga siendo “fétido”. El modelo, por consiguiente, debe aplicarse a todos los productos ya existentes y que están considerados como “fétidos”; a los existentes que van a ser modificados pero se prevé que tengan posibilidad de convertirse en “fétidos”; y a todos los nuevos que se vayan a desarrollar para que no se conviertan en “fétidos”.

Aplicación del modelo

La aplicación del modelo se puede hacer en distintas fases durante el proyecto software, pero la eficacia de las acciones variará según la fase del proyecto en que lo apliquemos:

- Si lo aplicamos en la fase de viabilidad será posible identificar muchas más acciones, se podrán estimar mejor los esfuerzos y planificar el proyecto adecuadamente, pero por el contrario puede ocurrir que los productos base cambien (por estar aún en desarrollo, en pruebas o en proyectos sin finalizar), los requisitos se modifiquen y seguramente los recursos que vamos a disponer sean desconocidos.
- Si lo aplicamos en la fase de diseño unitario, el proyecto y la base de diseño estarán muy estables y los recursos comprometidos y conocidos, pero, sin embargo, las posibles acciones preventivas serán más limitadas, el esfuerzo de modificación más grande si surgen problemas y más difícil la obtención de recursos apropiados.
- Si ya estamos en la fase de pruebas sólo se podrán identificar acciones para aumentar la eficacia de la detección de fallos, pero prácticamente ninguna actividad preventiva, siendo además muy cara la reparación de errores.

Una vez decidida la fase en la que vamos a aplicar el modelo, haremos una evaluación de cada uno de los factores, asignándole un valor de riesgo, y un peso según el impacto que produciría si se manifestara dicho riesgo (según tabla 1).

El resultado de evaluar todos los factores de riesgo determinará la posibilidad de producir un nuevo producto “fétido” o la desaparición de alguno ya identificado, y se mide como el porcentaje del valor de riesgo evaluado sobre el valor de riesgo máximo:

TABLA 1. CÁLCULO DEL VALOR DE RIESGO POR FACTOR		
• R = 0	Probabilidad baja de causar problemas	
• R = 1	Bajo riesgo de causar problemas	
• R = 2	Bastante probable de causar problemas	
• R = 3	Riesgo alto de causar problemas	
• R = 4	Riesgo muy alto causar problemas	
Factor de Riesgo	VR = R x P	Peso del Impacto
	• P = 0	SIN impacto, irrelevante
	• P = 1	Impacto MENOR en calidad o plan de tiempos
	• P = 2	Impacto MEDIO en calidad o plan de tiempos
	• P = 3	Impacto MAYOR
	• P = 4	Impacto CRÍTICO
Valor de Riesgo (VR)		

$$\text{VALOR DE RIESGO TOTAL} = \left[\frac{\sum (VR_i)}{\sum (VR_{\text{máximo}})} \right] * 100$$

Siendo VR_i el valor de riesgo evaluado para cada factor “i” y VR máximo el valor de sumar todos los Valores de Riesgo de todos los factores al máximo (es decir, VR máximo = 4*4* número de factores del modelo).

Para determinar qué actuación debemos llevar a cabo en el proyecto se aplicará la tabla 2

TABLA 2: PROBABILIDAD DE PRODUCIR UN PRODUCTO “FÉTIDO”

Valor de Riesgo Total	Probabilidad de producir un producto “fétido”
60% - 100%	ALTA Definitivamente “fétido”. Es crítico tomar acciones preventivas que se plasmarán en un plan de acción.
40% - 60%	MEDIA Posible “fétido”. Los riesgos mayores deben ser tratados y casi todos los factores de riesgo evaluados deben ser tenidos en cuenta. Se plasmará en un plan de acción.
30% - 40%	BAJA Se recomienda analizar los resultados con más cuidado para ver si los riesgos pueden ser asociados con causas similares y tratarlos.
Menos de 30%	MUY BAJA El diseño se puede considerar que está controlado.

(que puede ser adaptada para cada organización).

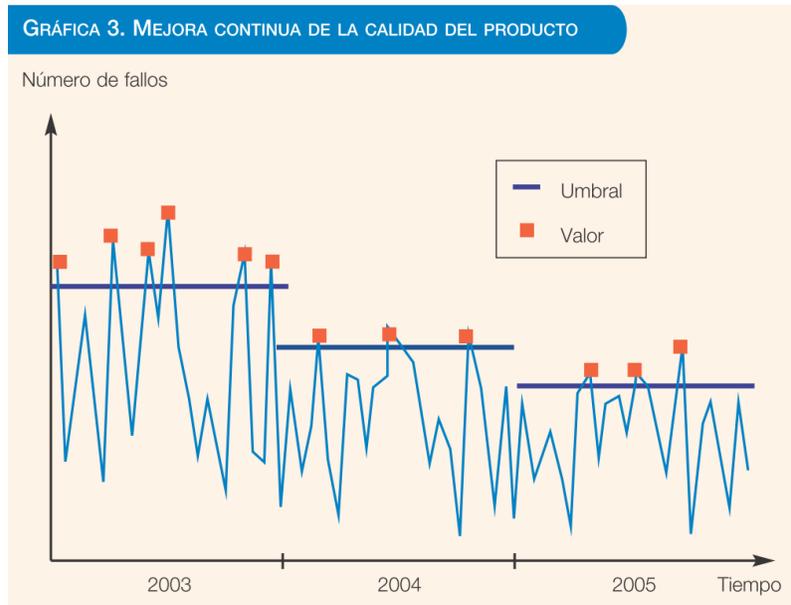
Resultados de la aplicación del modelo Al finalizar el proyecto se deben proporcionar las siguientes medidas de eficacia del modelo:

- Número de productos “fétidos” o potenciales “fétidos” que no se convirtieron en “fétidos”, es decir, las medidas adoptadas resultaron

apropiadas.

- Número de productos “fétidos” o potenciales “fétidos” que, a pesar de las medidas preventivas adoptadas, se convirtieron, finalmente, en “fétidos”.
- Número de productos “fétidos” que no fueron detectados como potenciales “fétidos”.

Impacto en los resultados y en la mejora continua El impacto de la aplicación de este modelo en los resultados es evidente en la gráfica 3 donde se ve que, año tras año, el número de fallos disminuye, lo que se traduce en ahorro de costes.



Por otro lado, debido a que las acciones preventivas tienen como efecto el que se produzcan menos errores, la consecuencia es que el nivel umbral, a partir del cual se aplica el modelo a los productos, va disminuyendo con la consiguiente mejora continua de la calidad de dichos productos. Se observa en la gráfica, que productos que no eran “fétidos” en 2003, en 2004 al bajar el nivel umbral sí lo son, y, por tanto,

se les aplica el modelo, volviendo a tener como consecuencia la disminución de nuevo del nivel umbral, y así sucesivamente en 2005 y posteriores, hasta conseguir que el nivel de fallos sea tan pequeño como sea rentable el disminuirlos.



personas



procesos



tecnologías

La especialización de las pruebas del software.

La especialización de las pruebas del software

La definición del ciclo de vida de desarrollo está presente en todas las organizaciones, con un mayor o menor nivel de metodología, procedimientos, soporte y herramientas. Los desarrolladores lo conocen y lo siguen de forma natural. Sin embargo, también está presente en todas las organizaciones el hecho de que el software en sí, por muy bien que haya sido construido, siempre produce fallos cuando está funcionando "en real", en el entorno y con los usuarios para los que fue concebido su uso.

Las pruebas tienen su propio ciclo de vida. En primer lugar, sería una pretensión excesivamente ambiciosa y, sobre todo, altamente ineficiente. Pero minimizar su aparición en los procesos críticos para el negocio sí es un objetivo alcanzable que se puede conseguir con un enfoque del problema adecuado.

Hay que empezar por reconocer que las pruebas tienen su propio ciclo de vida, y así debe estar definido en las organizaciones, con sus fases, sus entradas y salidas, sus tareas y sus productos.

La definición del ciclo de vida de pruebas no es suficiente en sí misma si no se determina su posicionamiento con respecto al ciclo de vida de desarrollo. Y es en ese punto donde se produce el gran cambio: ambos ciclos de vida, desarrollo y pruebas, transcurren en paralelo. Por dos razones fundamentales: la primera, porque las pruebas estáticas de documentación y de código ayudan a detectar problemas en la misma fase de desarrollo en la que son producidos, evitando así su traspaso a fases posteriores y el consecuente efecto multiplicador de los mismos. En segundo lugar, porque durante el diseño de las pruebas se

detectan también fallos, sin tener que llegar a la fase de ejecución de las mismas.

Esta anticipación del ciclo de vida de pruebas tiene como ventajas más significativas el hecho de que los errores más importantes se detectan antes, se previenen, no se actúa reactivamente, sin olvidar que éstos que se encuentran en etapas tempranas del proyecto se arreglan con un bajo coste. Asimismo, el diseño temprano de pruebas ayuda a mejorar la calidad y evita la multiplicación de fallos, ya que con el diseño de pruebas se encuentran errores, y los requisitos pueden sufrir cambios como consecuencia del diseño de pruebas.

Por último, hay que destacar que las pruebas estáticas de documentación y de código no sólo son un buen instrumento para detectar errores que pueden afectar a rendimiento, mantenibilidad, estándares, etc., sino que su uso sistemático contribuye a entender el uso de la metodología entre los desarrolladores y a prevenir futuros errores.

Aunque el ciclo de vida de pruebas puede variar de una organización a otra, normalmente está formado por las siguientes fases:

1. Enfoque de las pruebas, con un objetivo de definir la estrategia de pruebas para el proyecto o de ajustar el mismo a la estrategia general existente en la organización.
2. Planificación de las pruebas, incluyendo a caso en perfecta consonancia con la planificación del desarrollo del software.
3. Diseño de las pruebas, fase en la que se elabora un modelo de pruebas utilizando una combinación de las técnicas de diseño más apropiadas para el caso concreto.

4. Ejecución de las pruebas, con ayuda o no de herramientas de automatización de pruebas.
5. Cierre de las pruebas, con el fin de restaurar los entornos y preparar las lecciones aprendidas.

Los tipos de pruebas vienen determinados por el objetivo que se persigue.

Al igual que el desarrollo del software se hace paso a paso, las pruebas del mismo también se hacen en varios ataques. Hay una importante diferencia en ambos ciclos, así como en el desarrollo se parte de lo general (los requisitos, y los requisitos pueden sufrir cambios como consecuencia del diseño de pruebas), en las pruebas los pasos van en sentido contrario: se parte de las pruebas de la unidad más pequeña de software (los programas) para terminar con la prueba general del mismo en todo su ámbito (la prueba de aceptación de usuario).

Los tipos de pruebas se abordan consecutivamente, ampliando progresivamente su cobertura y variando sus objetivos. Una tipología de pruebas, ampliamente extendida, podría ser la que se presenta a continuación:

1. Pruebas unitarias, también llamadas de componentes.
2. Pruebas integradas de bajo nivel, orientadas a comprobar la correcta integración entre los distintos componentes.
3. Pruebas del sistema.
4. Pruebas integradas de alto nivel, orientadas a comprobar la correcta integración del sistema con el resto de sistemas de la organización.
5. Pruebas de rendimiento, carga y estrés.
6. Pruebas de aceptación de usuario.

ARTÍCULO

Autor: Luisa Morales Gómez-Tejedor Directora del Centro de Excelencia de Testing de Sopra PROFit, Vocal del Comité de Calidad del Software de la AEC

Publicado en la Revista Calidad Septiembre de 2007.

La definición del ciclo de vida de desarrollo está presente en todas las organizaciones, con un mayor o menor nivel de metodología, procedimientos, soporte y herramientas. Los desarrolladores lo conocen y lo siguen de forma natural. Sin embargo, también está presente en todas las organizaciones el hecho de que el software en sí, por muy bien que haya sido

construido, siempre produce fallos cuando está funcionando "en real", en el entorno y con los usuarios para los que fue concebido su uso.

Las pruebas tienen su propio ciclo de vida

Eliminar los fallos sería una pretensión excesivamente ambiciosa y, sobre todo, altamente ineficiente. Pero minimizar su aparición en los procesos críticos para el negocio sí es un objetivo alcanzable y que se puede conseguir con un enfoque del problema adecuado.

Hay que empezar por reconocer que las pruebas tienen su propio ciclo de vida, y así debe estar definido en las organizaciones, con sus fases, sus entradas y salidas, sus tareas y sus productos.

La definición del ciclo de vida de pruebas no es suficiente en sí misma si no se determina su posicionamiento con respecto al ciclo de vida de desarrollo. Y es en ese punto donde se produce el gran cambio: ambos ciclos de vida, desarrollo y pruebas, transcurren en paralelo. Por dos razones fundamentales: la primera, porque las pruebas estáticas de documentación y de código ayudan a detectar problemas en la misma fase de desarrollo en la que son producidos, evitando así su traspaso a fases posteriores y el consecuente efecto multiplicador de los mismos. En segundo lugar, porque durante el diseño de las pruebas se detectan también fallos, sin tener que llegar a la fase de ejecución de las mismas.

Esta anticipación del ciclo de vida de pruebas tiene como ventajas más significativas el hecho de que los errores más importantes se detectan antes, se previenen, no se actúa reactivamente, sin olvidar que éstos que se encuentran en etapas tempranas del proyecto se arreglan con un bajo coste. Asimismo, el diseño temprano de pruebas ayuda a mejorar la calidad y evita la multiplicación de fallos, ya que con el diseño de pruebas se encuentran errores, y los requisitos

pueden sufrir cambios como consecuencia del diseño de pruebas.

Por último, hay que destacar que las pruebas estáticas de documentación y de código no sólo son un buen instrumento para detectar errores que pueden afectar a rendimiento, mantenibilidad, estándares, etc., sino que su uso sistemático contribuye a extender el uso de la metodología entre los desarrolladores y a prevenir futuros errores.

Aunque el ciclo de vida de pruebas puede variar de una organización a otra, normalmente está formado por las siguientes fases:

1. Enfoque de las pruebas, con un objetivo de definir la estrategia de pruebas para el proyecto o de adaptar al mismo la estrategia general existente en la organización.
2. Planificación de las pruebas, llevándose a cabo en perfecta consonancia con la planificación del desarrollo del software.
3. Diseño de las pruebas, fase en la que se elabora un modelo de pruebas utilizando una combinación de las técnicas de diseño más apropiadas para el caso concreto.
4. Ejecución de las pruebas, con ayuda o no de herramientas de automatización de pruebas.
5. Cierre de las pruebas, con el fin de restaurar los entornos y preparar las lecciones aprendidas.

Los tipos de pruebas vienen determinados por el objetivo que se persigue

Al igual que el desarrollo del software se hace paso a paso, las pruebas del mismo también se hacen en varias etapas. Hay una importante diferencia en ambos ciclos, así como en el desarrollo se parte de lo general (los requisitos, el diseño conceptual) hasta llegar a la mínima parte del software que soporta el sistema (los programas), en las pruebas los pasos van en sentido contrario: se parte de las pruebas de la unidad más pequeña de software (los programas) para terminar con la prueba general del mismo en toda su amplitud (la prueba de aceptación de usuario).

Los tipos de pruebas se abordan consecutivamente, ampliando progresivamente su cobertura y variando su objetivo. Una tipología de pruebas, ampliamente extendida, podría ser la que se presenta a continuación:

1. Pruebas unitarias, también llamadas de componentes.
2. Pruebas integradas de bajo nivel, orientadas a comprobar la correcta integración entre los distintos componentes.
3. Pruebas del sistema.
4. Pruebas integradas de alto nivel, orientadas a comprobar la correcta integración del sistema con el resto de sistemas de la organización.
5. Pruebas de rendimiento, carga y estrés.
6. Pruebas de aceptación de usuario.

Las técnicas de diseño de pruebas tienen la clave del éxito

Un buen ciclo de vida de pruebas, por el que pasan los distintos niveles de pruebas, necesita, además, un buen diseño del modelo de pruebas para garantizar el éxito de las mismas. ¿Cómo sabemos que un diseño de pruebas es bueno? Lo sabemos por dos características: su eficacia y su eficiencia. Es decir, que el diseño tiene que ir orientado a encontrar el mayor número de fallos posibles en el menor tiempo. En todo proyecto, la curva de aparición de fallos tiene la forma de una “S”, es decir, al principio se encuentran pocos errores, luego hay una curva ascendente hasta que llega un momento en el tiempo en el que, con mucho esfuerzo, se encuentran pocos errores.

¿Cuál es la mejor técnica de diseño? No existe la mejor técnica, sino la más adecuada a cada caso concreto, y, normalmente, la combinación de varias técnicas proporciona un diseño de pruebas bueno para el objetivo perseguido. Podemos agrupar las técnicas en tres grandes bloques:

1. Técnicas de caja blanca, basadas en la estructura del software: cobertura de sentencias, decisiones, módulos/componentes, puntos de menú, etc.
2. Técnicas de caja negra, basadas en las especificaciones del sistema: rangos de valores equivalentes, valores limítrofes, transición de estados, tablas de decisión, casos de uso, etc.
3. Técnicas de testing rápido de software, basadas en la experiencia del equipo de pruebas: testing exploratorio, "error guessing", "lateral testing", etc.

Las técnicas mayoritariamente extendidas son las pertenecientes a los dos primeros grupos, caja blanca y caja negra, siendo el tercer grupo un gran desconocido, aunque muchas veces es utilizado sin tener la plena conciencia de ello.

El "testing rápido" de software es la manera más rápida y menos cara para cumplir el objetivo de probar un software.

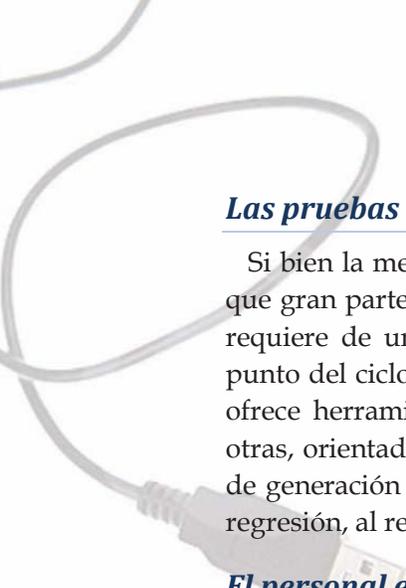
Tradicionalmente, el testing se ha orientado a obtener las respuestas correctas, a través de pruebas minuciosamente documentadas, controladas y repetibles, pero, hay tantas preguntas que responder, que tenemos que considerar las ventajas de obtener las respuestas correctas, frente al peligro de simplemente no hacer suficientes preguntas. El testing rápido de software es adecuado cuando hay poca o ninguna documentación de partida, poco tiempo disponible, pocos recursos y poco conocimiento de la aplicación, o el objetivo es evaluar las áreas con más puntos débiles del sistema.

También es útil para complementar el "testing formal" y chequear los procesos de pruebas. El factor clave es la experiencia del equipo de pruebas.

Algunas de las técnicas más conocidas pertenecientes a este grupo son:

1. El testing exploratorio, basado en un mapa de pruebas que contiene los objetivos / condiciones de las mismas. Las pruebas se realizan siguiendo los caminos lógicos que marca el propio software. El resultado de las pruebas que se diseñan y ejecutan influenciará las siguientes pruebas que se decida diseñar y ejecutar. Podemos decir que es una técnica simultánea, ya que el conocimiento, diseño, ejecución y registro de las pruebas se hace en la misma fase. Los testers aprenden el manejo del software según lo prueban.
2. "Error guessing". Esta técnica suple o complementa a las técnicas sistemáticas, es probablemente la más usada. Puede encontrar defectos que otras técnicas más sistemáticas podrían dejar pasar. Busca los posibles defectos usando intuición, experiencia, el pasado de otros defectos / fallos encontrados y brainstorming.
3. Otras técnicas pertenecientes a este grupo son el "lateral testing", que utiliza el pensamiento lateral para el diseño de las pruebas; "la prueba del mono", que se enfrenta a una transacción sin seguir ninguna lógica predeterminada, etc.

Hay que tener presente que una técnica exitosa en un proyecto anterior no tiene por qué ser la más adecuada para el proyecto actual, por lo que hay que analizar el contexto y seleccionar la combinación de técnicas más adecuada para el diseño de pruebas en un caso concreto. A modo de ejemplo, si nos enfrentamos a una exigencia de prueba en profundidad de todas las reglas de negocio, o de gran cantidad de combinaciones de factores, las tablas de decisión serán una técnica muy adecuada, mientras que si hay una alta criticidad de navegación, la técnica de transición de estados puede ser la más conveniente.



Las pruebas requieren herramientas específicas

Si bien la mejor “herramienta” es un buen proceso de pruebas definido y aplicado, es cierto que gran parte de las pruebas se facilita, e incluso se posibilita, con el uso de herramientas. Se requiere de un buen conocimiento del proceso y de la instalación para determinar en qué punto del ciclo de vida va a aportar más valor el uso de herramientas. En general, el mercado ofrece herramientas para gran parte del ciclo de vida de pruebas: hay herramientas, entre otras, orientadas a la gestión y diseño de las pruebas, a la preparación de las mismas a través de generación y selección de datos, a la ejecución de pruebas, muy útiles para las pruebas de regresión, al rendimiento de los sistemas, al análisis de código, etc.

El personal especializado es la base del proceso

El factor humano es fundamental en la especialización de las pruebas de software. Un personal formado en el método, las técnicas y las herramientas es la clave para obtener resultados tangibles desde el primer momento. Entre las distintas opciones de formación a las que se puede optar, destaca la certificación en pruebas del software otorgada por el ISEB/ISTQB.

El ISEB (Information Systems Examination Board) forma parte de la British Computer Society, organismo independiente, líder en el campo de los sistemas de información, que, entre otros, establece los estándares técnicos y éticos para la industria informática. El ISEB proporciona certificaciones reconocidas por la industria que miden la competencia, habilidad y performance en muchas áreas de los sistemas de información, con el objetivo de establecer estándares de la industria y promover el desarrollo de las carreras profesionales.

El testing del software es una de las áreas más recientes de certificación del ISEB. Esta certificación es otorgada por el ISTQB (International Software Testing Qualification Board), asociación de representantes de un gran número de países que otorgan una certificación en pruebas del software reconocida en todo el mundo.

Este tipo de formación se complementa con la específica en herramientas para permitir al personal de testing la instalación, parametrización, adaptación y uso de las mismas.

La especialización de las pruebas mejora el ratio de incidencias en producción

Las empresas que han apostado por profesionalizar las pruebas a través de la creación de equipos independientes de pruebas han obtenido resultados altamente satisfactorios, mejorado significativamente el ratio de incidencias por Punto Función de las aplicaciones en producción.

Los beneficios que destacan, entre otros, son una mayor fiabilidad debido al aumento de la confianza tanto del usuario final como de otros equipos externos e internos implicados, una mejora de costes y plazos, con una repercusión directa en el desarrollo de proyectos; y un incremento de la calidad percibida materializada en la satisfacción de los usuarios finales además de las personas involucradas en el desarrollo y pruebas de los proyectos, que quedan motivadas por el trabajo bien hecho.

Modelos

Desde la aparición de la ISO 9000 el sector de las TIC se ha caracterizado por un abanico increíble de metodologías, modelos y marcos de referencia que desde la absoluta globalidad de la problemática en el Software hasta especializaciones en ámbitos como el desarrollo o las pruebas han ido apareciendo.

Hoy en día CMMI, ITIL, o Cobit conviven con naturalidad, en algunos casos demasiada, dentro de las organizaciones.

Desde el año 2000, la norma ISO 9000 promueve la adopción de un enfoque basado en procesos cuando se desarrolla, implementa y mejora un sistema de gestión de la calidad. El enfoque basado en procesos está reflejado en la estructura de la norma ISO 9004:2000 “Sistemas de gestión de la calidad. Directrices para la mejora del desempeño”, y también en la norma ISO 9001:2000.

La estructura del sistema de gestión de la calidad basado en procesos, se muestra a continuación, esquemáticamente, en la Figura 1.

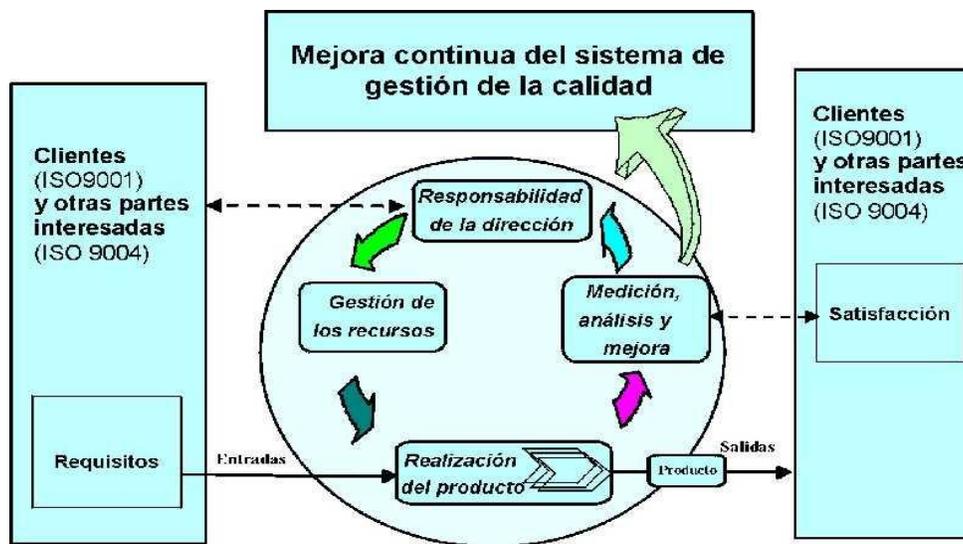


Figura 1 - Modelo de un sistema de gestión de la calidad basado en procesos (tomado de la Norma ISO 9000:2000)

Es importante comprender qué se entiende por un proceso y cómo los procesos pueden interaccionar dentro de un sistema, así como definir el método para gestionar esos procesos.

Atendiendo a la definición de proceso, los elementos de entrada (entrada) y los resultados (salida) pueden ser tangibles o intangibles, por ejemplo pueden ser equipos, materiales,

componentes, información y recursos financieros, entre otros.

Para realizar las actividades dentro del proceso hay que asignar recursos apropiados y se debe emplear un sistema de medición para reunir información y datos con el fin de analizar el desempeño del proceso y las características de entrada y de salida.

En la Figura 2, se describe una representación del esquema que debe presentar cualquier proceso:

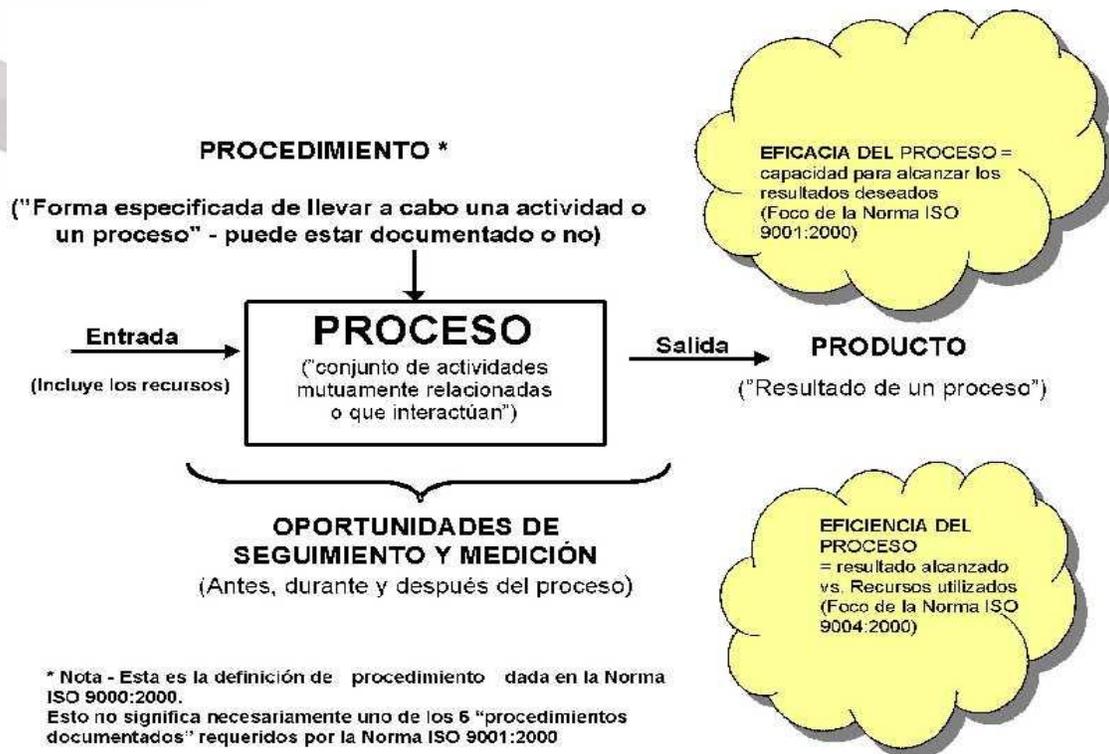


Figura 2 - Representación esquemática de un proceso

Las direcciones de las empresas deben hacer énfasis especial en identificar, implementar, gestionar y mejorar continuamente la eficacia de los procesos que son necesarios para el Sistema de Aseguramiento de la Calidad, y para gestionar las interacciones de esos procesos con el fin de alcanzar los objetivos de la organización. Para ello se recomienda una evaluación de la eficiencia de los procesos.

La eficiencia del proceso puede evaluarse a través de los procesos de revisión internos o externos y valorarse en una escala de madurez. Estas escalas se dividen en grados de madurez desde un "sistema informal" hasta "el de mejor desempeño en su clase". Una ventaja de este enfoque es que los resultados pueden ser documentados y seguidos en el tiempo hasta alcanzar las metas de mejora.

La dirección de las empresas debe asumir la responsabilidad de la planificación de la calidad de toda la organización. Esta planificación tiene el objeto de satisfacer los siguientes requisitos:

- Identificar los procesos clave y de soporte existentes en la compañía.
- Determinar la secuencia e interacción de estos procesos.
- Determinar los métodos y criterios necesarios para asegurar el funcionamiento efectivo y el control de los procesos.

- 
- Asegurar la disponibilidad de la información necesaria para apoyar el funcionamiento y el seguimiento de los procesos.
 - Llevar a cabo la medición, seguimiento y análisis de estos procesos, así como la implantación de las acciones necesarias para lograr los resultados esperados y la mejora continua.
 - Gestionar estos procesos

Para ello es muy importante tomar las referencias que en el mercado existen respecto a lo que podemos denominar modelos formales como pueden ser:

CMMI
ITIL
Cobit
eSCM
TOGAF

Principalmente, estos modelos pueden ayudar a la organización a delimitar el alcance de su actuación así como a centrar las ideas de modelización de Procesos dentro de la organización. Estos procesos serán claves para la implantación de un sistema de aseguramiento de la calidad robusto.

Artículos presentados en esta selección:

- [La importancia de la certificación: De ISO a SW-CMM.](#)
- [Estado del Arte ISO/IEC 15504 Estado del Arte en la ISO/IEC 15504.](#)
- [Cómo implantar CMMI y no morir en el intento](#)
- [CMMI DEV Diferencias entre nivel 2 y nivel 3 y una estrategia de implantación](#)



personas



procesos



tecnologías

La importancia de la certificación: De ISO a SW-CMM.

Artículo

La importancia de la certificación: de ISO a SW-CMM

Actualmente todo el mundo apuesta por la calidad, o a menos, no hay nadie que se atreva a decir que no trabaja con calidad o que no persigue la mejora de su calidad. En nuestro ecosistema profesional, sería un eslabón perdido... Desafortunadamente, y a pesar del aparente generalizado interés por este concepto, la situación real es muy diferente. La realidad, tozuda, se empeña todos los días en demostrar que existe aún una clara falta de entendimiento de lo que la calidad, y la falta de ella, significa e implica.

En España, y en pocos años, se ha producido una rápida y positiva evolución en el ambiente de las organizaciones y compañías de tecnologías de la información. Aun así, todavía algunas de ellas no se han planteado seriamente la definición de calidad o peor aún, no han mostrado la más mínima preocupación. Como se suele decir: "la calidad les persigue, pero ellos corren más". De todas formas, y aunque sería obvio recordar todas las bondades de los sistemas de calidad, si alguien a estas alturas aún dudase de sus beneficios, más allá de su empleo como reclamo publicitario, tendría un serio problema.

Con este panorama como marco de referencia, certificaciones como ISO o modelos como CMM han venido a concienciar, guiar e impulsar iniciativas, que anteriormente —y en muchas organizaciones— dependían del heroísmo y empuje de sólo unos pocos. Su prestigio, o la pérdida de prestigio o posición por su no obtención, han servido de palanca en muchas ocasiones.

A partir de los años noventa en nuestro país vivimos la oleada de la familia de normas ISO 9000, que sirvió de catalizador para muchos, hasta el punto que aún hoy es la certificación más generalizada y predominante en nuestro mercado, con más de 350.000 organizaciones certificadas en todo el mundo. En España contamos con 28.700, y de ellas, 400 en nuestro sector. ISO ha pasado de ser inicialmente un elemento diferenciador a ser considerada casi un estándar sine qua non. Hoy, organizaciones interesadas por la mejora continua en la calidad de sus productos y servicios sienten la necesidad de ir más allá, buscando la diferenciación y la evolución de sus sistemas ya implantados. Es, en este ambiente, donde está surgiendo una nueva oleada de interés por modelos como SW-CMM y CMMI, o herramientas como Six-Sigma, ISO 15504 y otros.

Tal vez el más conocido de entre ellos sea SW-CMM, tanto por sus ya muchos años en el

Autor Ulises Arranz Director Asociado de Coritel, empresa miembro del Comité de Software de la AEC

Publicado en la Revista Calidad – Abril-Mayo 2004

Actualmente todo el mundo apuesta por la calidad, o al menos, no hay nadie que se atreva a decir que no trabaja con calidad o que no persigue la mejora de su calidad. En nuestro darwinista ecosistema profesional, sería un eslabón perdido... Desafortunadamente, y a pesar del aparentemente generalizado interés por este concepto, la situación real es muy diferente. La realidad, tozuda, se empeña todos los días en demostrar que existe

e impulsar iniciativas, que anteriormente —y en muchas organizaciones— dependían del heroísmo y empuje de sólo unos pocos. Su prestigio, o la pérdida de prestigio o posición por su no obtención, han servido de palanca en muchas ocasiones.

A partir de los años noventa en nuestro país vivimos la oleada de la familia de normas ISO 9000, que sirvió de catalizador para muchos, hasta el punto que aún hoy es la certificación más generalizada y predominante en nuestro mercado, con más de 350.000 organizaciones certificadas en todo el mundo. En España contamos con 28.700, y de ellas, 400 en nuestro sector. ISO ha pasado de ser inicialmente un elemento diferenciador a ser considerada casi un estándar sine qua non. Hoy, organizaciones interesadas por la mejora continua en la calidad de sus productos y servicios sienten la necesidad de ir más allá, buscando la diferenciación y la



194 calidad • abril-mayo 2004

mercado como por su prestigio y amplia extensión, hasta convertirse en casi un estándar internacional. SW-CMM es un modelo de procesos organizado en cinco niveles y enfocado a la mejora evolutiva de la organización. Fue desarrollado a finales de los ochenta por el Software Engineering Institute de la Universidad Carnegie-Mellon a petición del Departamento de Defensa de EE.UU.

No obstante, y aunque el interés por estos modelos exista, muy pocas organizaciones tienen la oportunidad y capacidad de desarrollarlos y aplicarlos. Como muestra, dos ejemplos: en España, sólo tres organizaciones están reconocidas oficialmente con algún nivel SW-CMM, y a nivel mundial, sólo el 6% de las organizaciones que aplican SW-CMM han obtenido su máximo nivel.

En una organización como Coritel (Grupo Accenture) se ha disfrutado del raro privilegio de vivir ambas oleadas: la familia ISO 9000 desde 1996 y la nueva, SW-CMM, desde 1999. Del camino que se ha seguido, corto para los estándares del SEI hasta alcanzar el máximo nivel posible SW-CMM5 – el pasado mes de octubre de 2003 – se pueden obtener muchas lecciones y beneficios. Pero una cosa es segura, no se habría alcanzado esos niveles de excelencia, prestigio y reconocimiento, sin la guía y ayuda que han supuesto ISO 9001:2000 y SW-CMM.

Claves, ambas, en la implantación de la mejora continua como un hábito de la organización, marcando una agenda, un calendario y unos retos que han empujado una evolución constante.



Estado del Arte ISO/IEC 15504 Estado del Arte en la ISO/IEC 15504.



Autor: Luis Redondo López
Miembro del comité de Software de la AEC
Director de calidad y nuevos servicios de la compañía MÉTODOS Y TECNOLOGÍA

Publicado en la Revista Calidad - Septiembre 2004

Este artículo detalla el estado actual del estándar ISO/IEC 15504, en lo que se refiere a la elaboración del estándar y su aplicación en proyectos reales. El estándar 15504 está actualmente en proceso de convertirse en un estándar internacional y lo que aquí pretendemos es recopilar información, detallando en qué ha consistido la revisión de la estructura de la documentación, cuál es el progreso de la elaboración de la documentación y desvelando parte del contenido del futuro estándar. Por otro

lado, también reflejamos en qué medida se está aplicando este estándar, es decir, los principales sectores empresariales, qué procesos son los que se suelen valorar (se prefiere emplear el término "valorar" frente al de "evaluar" porque queremos resaltar el aspecto cuantitativo del resultado) y los correspondientes niveles de capacidad alcanzados.

Palabras clave: SPICE, valoración, assessment, proceso, estándar, 15504, ISO, IEC, norma, SAS, Automotive, ESA.

Introducción

"Reducir costes y mejorar la calidad evitando problemas", éste podría ser un objetivo de cualquier organización dentro de sus planes de calidad. El estándar 15504 es una herramienta que ayuda a conseguirlo.

La ISO/IEC TR 15504, conocida como SPICE (Software Process Improvement and Capability Determination), es un marco de valoración de procesos software, el cual se puede emplear por aquellas organizaciones involucradas en la planificación, gestión, monitorización, control y mejora de la adquisición, suministro, desarrollo, operación, evolución y soporte de software.

Pero la ISO/IEC TR 15504 no es sólo un método de valoración (assessment), sino que está diseñada para facilitar una aproximación común para realizar valoraciones de procesos, haciendo posible comparaciones entre los resultados de las mismas. Estos resultados se pueden basar en diferentes modelos de valoración, siempre que sean compatibles con el estándar, y métodos de valoración, siempre que cumplan los requisitos del estándar (ISO/IEC 15504-3).

En este artículo, vamos exponer en qué situación se encuentra la elaboración del estándar 15504 (SPICE) y una visión general de las principales partes de la misma.

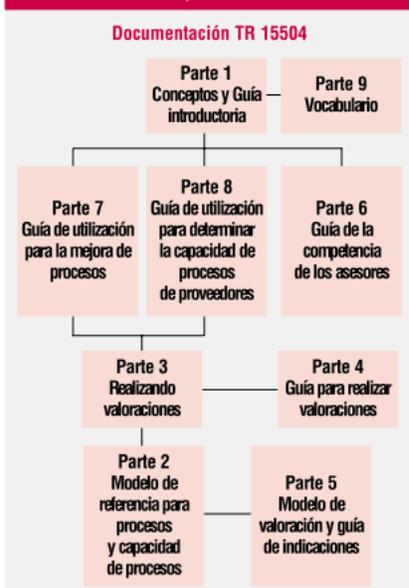
Asimismo, se va a facilitar ciertos datos de mercado relativos al grado de aplicación de valoraciones SPICE a nivel mundial como: países donde se emplea frecuentemente, principales sectores donde se está aplicando y los procesos que más preocupan a las organizaciones.

Del informe técnico al estándar internacional

La ISO/IEC TR 15504, publicada en 1998, está estructurada en nueve partes, de la parte 1 a la parte 9. Para la realización de valoraciones conformes con la 15504, las partes 2, 3, 4 y 5 son las que contienen la información más relevante.

La parte 2 (normativa) detalla el modelo de referencia, el cual es un modelo bidimensional para describir los procesos y sus capacidades, basándose en la ISO 12207, así como los requisitos que debe satisfacer cualquier modelo de valoración. La parte 5 (informativa) contiene el modelo de valoración, el cual incluye toda la parte 2 ampliado con los indicadores de realización (prácticas base, productos de trabajo y características de los productos de trabajo) y los indicadores de capacidad (prácticas de gestión junto con las características de infraestructura y de recursos). La parte 3 (normativa) define los requisitos de cualquier método

Figura 1: Estructura de la documentación de la ISO/IEC TR 15504



de valoración conforme a la 15504 con objeto de que sea repetible, fiable y consistente; para ello incluye las entradas necesarias, responsabilidades, el proceso, la notificación y el registro de las salidas de una valoración. Finalmente, la parte 4 (informativa) facilita una guía para llevar a cabo valoraciones.

Las partes 7 y 8 (informativas) contienen detalles de cómo emplear los resultados de una valoración para la Mejora de Procesos SW (SPI, Software Process Improvement) y para la determinación de capacidades (CE, Capability dEtermination), respectivamente. Finalmente, la parte 6 (informativa) contiene detalles sobre las habilidades y competencias de los asesores SPICE.

El **Modelo de Referencia** es bidimensional:

- Por un lado tenemos la dimensión de los procesos, caracterizado por los propósitos de los procesos que son objetivos medibles.
- Por otro lado tenemos la dimensión de la capacidad de los procesos, caracterizado por los atributos de los procesos organizados en niveles de capacidad, aplicables a cualquier proceso.

La futura IS 15504 (International Standard) estará compuesta por cinco partes:

- Parte 1: Conceptos y vocabulario.
- Parte 2: Realizando una valoración.
- Parte 3: Guía para realizar valoraciones.
- Parte 4: Guía para emplear los resultados de valoraciones.
- Parte 5: Un ejemplo de modelo de valoración.

Los **principales cambios** corresponden a:

- Se ha eliminado “software” del título, por lo que se reconocerá a la 15504 como un

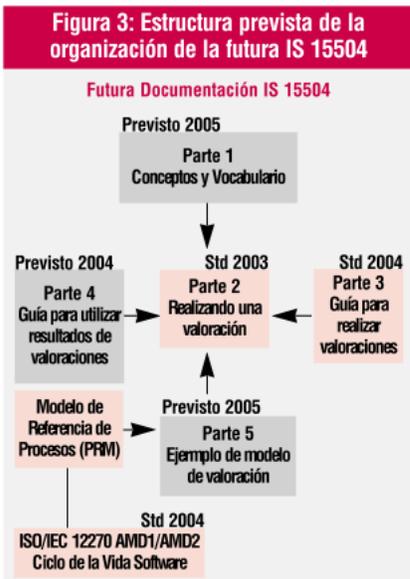
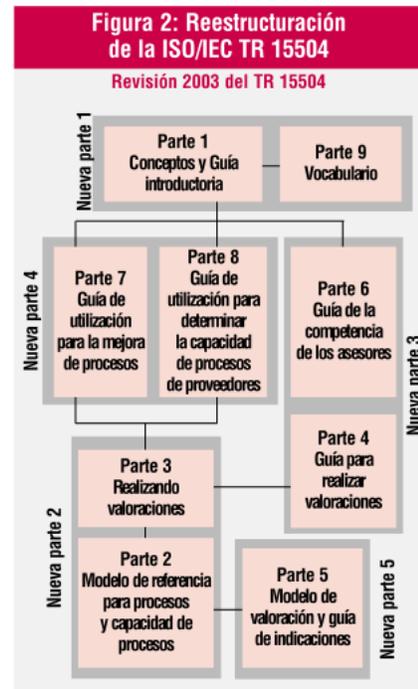
método de valoración aplicable a otros dominios distintos del software.

- La documentación se ha revisado y reestructurado, reduciendo las nueve partes en cinco. La Parte 2 es la que únicamente contendrá todos los requisitos.
- La dimensión de los procesos se ha reemplazado por el concepto de Modelo de Referencia de Procesos (PRM, Process Reference Model). De esta forma, una valoración se podrá realizar empleando cualquier PRM.

Por defecto, el futuro estándar recomendará la utilización conjunta de la ISO 12207 y la 15504 para software.

La correspondencia entre las partes de la ISO/IEC TR y el nuevo estándar 15504 son:

- Las partes 1 y 9 se han incluido en la nueva parte 1.
- Las partes 2 y 3 se han incluido en la nueva parte 2, excepto que la dimensión de los procesos se reemplaza por la ISO 12207 AMD1/AMD2.
- Las partes 7 y 8 se han incluido en la nueva parte 4.
- La parte 5 se traslada a la nueva parte 5.



En la figura 3 mostramos gráficamente cómo se espera que quede formalmente estructurada la documentación de la ISO 15504.

La dimensión de los procesos

En la dimensión de los procesos, los principales procesos involucrados en la producción de software están agrupados en tres grandes áreas: primarios, de soporte y de organización. A su vez, estos grupos contienen cinco grandes categorías de procesos: procesos de cliente-suministrador (identificados como CUS), procesos de ingeniería (identificados como ENG), procesos de soporte (identificados como SUP), procesos de gestión (identificados como MAN) y procesos de organización (identificados como ORG).

Esta estructuración de los procesos es consistente con la ISO 12207. De hecho, con las actualizaciones de la ISO 12207, lo que se pretende es aproximar la 12207 a la vigente 15504-2, de tal forma que en la futura IS 15504, tendremos que emplear conjuntamente la actualizada 12207 y la 15504 porque la actual 15504-2 habrá desaparecido (ver fig. 4).

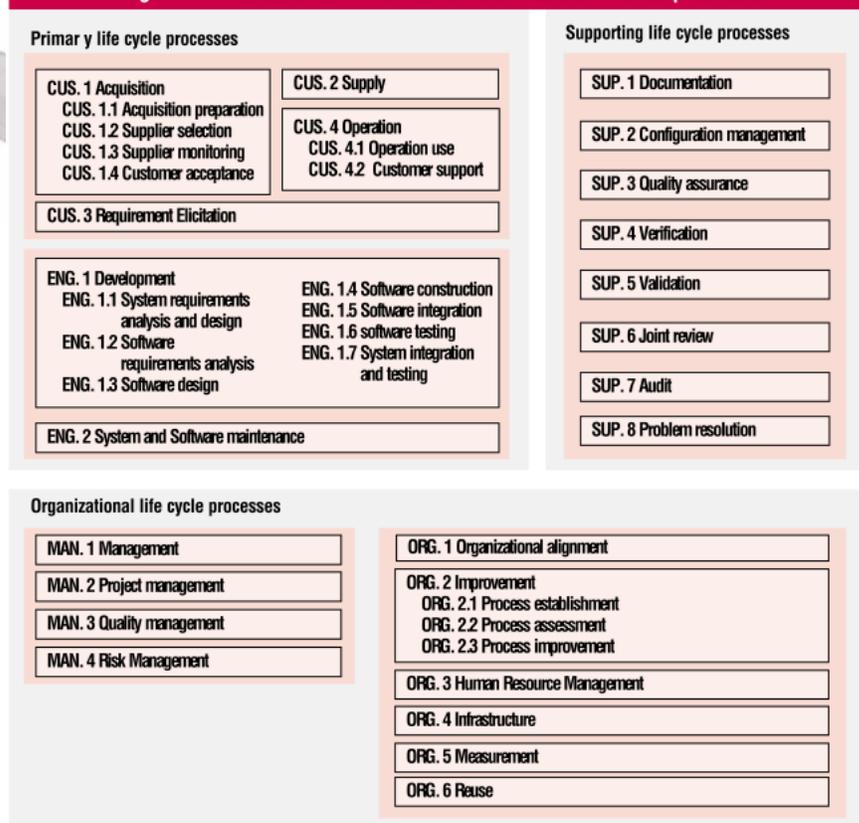
Los procesos están descritos mediante los siguientes elementos:

- Identificador de proceso: compuesto por la categoría del proceso y un número secuencial, por ejemplo: CUS.1.
- Nombre de proceso: frase descriptiva del proceso.
- Tipo de proceso: hay cinco tipos de proceso (básico, extendido, nuevo, componente y

componente extendido) según si es idéntico, ampliado, nuevo, parte o parte ampliada de un proceso de la 12207.

- Propósito de proceso: un párrafo describiendo a alto nivel los objetivos del proceso.
- Salidas del proceso: los resultados observables de una implementación con éxito del proceso.
- Observaciones del proceso: lista opcional de aclaraciones relativas al proceso.

Figura 4: Procesos identificados en la dimensión de los procesos



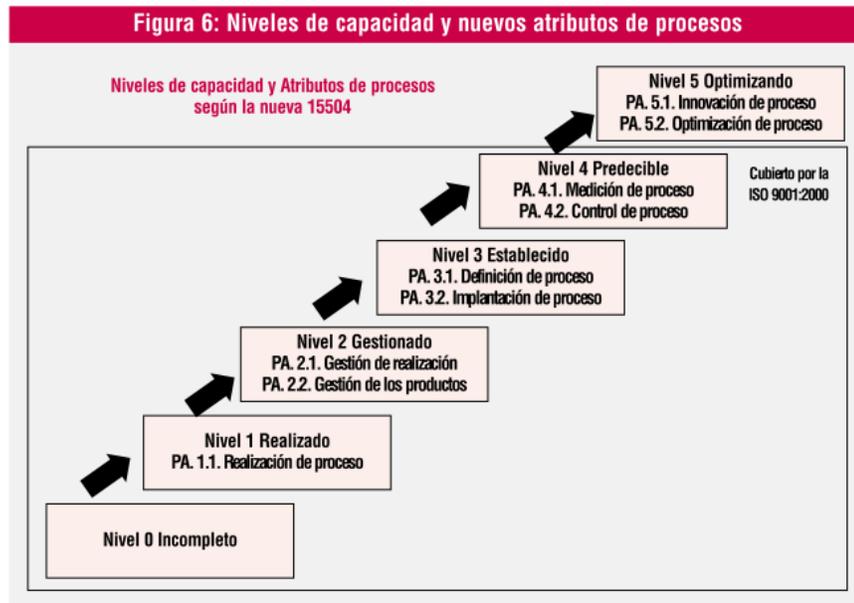
La nueva dimensión de capacidad

La capacidad de un proceso se define en una escala ordinal de seis puntos. La escala refleja el incremento de capacidad de un proceso desde su nivel más bajo '0' hasta su nivel más alto '5'. La propia escala pone de manifiesto la ruta para ir mejorando un proceso.

- Nivel 0 "Proceso incompleto": el proceso no está implementado o falla al conseguir el propósito. Generalmente, la situación es caótica.
- Nivel 1 "Proceso realizado": el proceso implementado consigue el propósito del proceso. Nos enfocamos en los productos y no en cómo se consiguen éstos. Generalmente, hay héroes en la organización que hacen posible que se consigan productos.
- Nivel 2 "Proceso gestionado": el proceso realizado ahora se implementa de forma gestionada (planificado, monitorizado y ajustado) y sus correspondientes productos resultantes son establecidos, controlados y mantenidos.
- Nivel 3 "Proceso establecido": el proceso gestionado está ahora implementado utilizando un proceso definido que es capaz de conseguir los resultados del proceso. Un proceso definido será un proceso derivado de un proceso estándar. A su vez, un

proceso estándar es un proceso de la organización derivado de buenas prácticas de ingeniería del software.

- Nivel 4 “Proceso predecible”: el proceso establecido ahora opera dentro de unos límites definidos para conseguir los resultados del proceso. La realización del proceso se entiende en términos



cuantitativos y permite a la organización identificar las causas de las variaciones en los procesos para poder llevar a cabo acciones correctoras.

Tabla 1: Escala de valores aplicable a cada atributo

Abr.	Nombre	%	Comentarios
N	No conseguido	[0, 15]	No hay evidencia o es escasa.
P	Parcialmente conseguidos	(15,50)	Hay evidencia de sistematización, pero algunos aspectos son impredecibles.
L	Ampliamente (Largely) conseguido	(50, 85)	Hay evidencia de sistematización, pero hay variabilidad impredecible.
F	Totalmente (Fully) conseguido	(85, 100)	Hay evidencia de una completa y sistemática aproximación para conseguir el atributo. No se identifican debilidades.

- Nivel 5 “Proceso de optimización”: el proceso predecible está continuamente mejorado para satisfacer objetivos relevantes de negocio, tanto actuales como proyectados. Los objetivos son cuantitativos y alineados con los objetivos del negocio. Además este nivel implica el llevar a cabo proyectos pilotos de ideas y tecnologías innovadoras con objeto de ir mejorando los procesos de forma continua.

Tabla 2: Nivel alcanzado según valoración atributos de los procesos

Atributos de proceso	Niveles de capacidad				
	1	2	3	4	5
1.1	L/F	F	F	F	F
2.1		L/F	F	F	F
2.2		L/F	F	F	F
3.1			L/F	F	F
3.2			L/F	F	F
4.1				L/F	F
4.2				L/F	F
5.1					L/F
5.2					L/F

Cada nivel tiene asociados unos atributos de procesos. Un atributo de proceso es una característica medible de la capacidad de cualquier un proceso. Disponemos de nueve atributos de procesos distribuidos en los distintos niveles. Cada atributo se mide de acuerdo a una escala. Esto da un gran nivel de detalle y nos permite poder aplicar acciones de mejora en partes específicas de cualquier proceso.

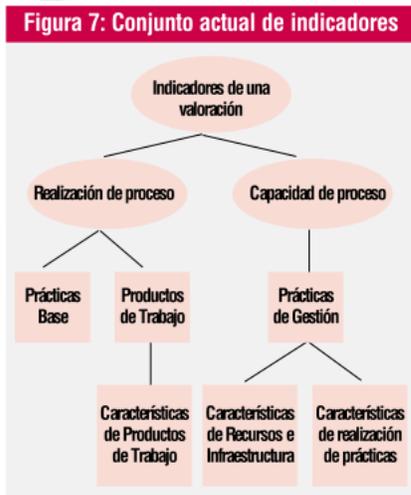
La valoración de procesos

Cada atributo de proceso se evalúa de acuerdo a una escala ordinal de cuatro valores.

El nivel alcanzado por un proceso se deriva de las valoraciones de los atributos de los procesos de acuerdo a la siguiente tabla (ver tabla 2).

Para valorar los atributos de los procesos, tendremos que valorar las evidencias objetivas que identifiquemos en la valoración de los procesos. Estas evidencias objetivas son las

que se conocen como indicadores. Los indicadores, tal y como están definidos en la 15504-5, son de dos tipos: de realización de proceso (correspondientes al nivel 1) y de capacidad de proceso (correspondientes al resto de los niveles de capacidad).



Los indicadores de un modelo de valoración representan diferentes tipos de evidencias objetivas que se podrían encontrar en un proceso y, por tanto, se podrían emplear para juzgar el cumplimiento de la realización o de una capacidad de un proceso. Todas las referencias a las evidencias objetivas encontradas deberemos registrarlas para poder hacer una trazabilidad de los resultados.

Valoraciones realizadas

Hasta la fecha se han realizado más de 4.000 valoraciones en todo mundo, distribuidas en 45 países.

Los países donde principalmente se realizan valoraciones SPICE son: Alemania, Francia, Inglaterra, Japón, Corea, Australia y Brasil.

Los sectores donde principalmente se aplica SPICE, ordenados de mayor a menor grado de intensidad, son: Espacio, Automoción, Defensa, Aviónica y Medicina.

Como hecho relevante, debemos mencionar también que Microsoft ha adoptado SPICE como base en su Microsoft Readiness Framework.

Los procesos que se han valorado con una mayor frecuencia son los relativos a la gestión de proyectos, al aseguramiento de calidad, procesos de desarrollo software, gestión de configuración software, gestión del riesgo y elicitación de requisitos de los usuarios.

Estos procesos, de acuerdo a la nomenclatura de la Dimensión de los Procesos (ver sección 3), son:

- MAN.2 - Project Management
- SUP.3 - Quality Assurance
- ENG.1.2 - Software Requirements
- ENG.1.3 - Software Design
- ENG.1.4 - Software Construction
- ENG.1.5 - Software Integration
- ENG.1.6 - Software Testing
- SUP.2 - Configuration Management
- MAN.4 - Risk Management
- CUS.3 - Requirements Elicitation

Por lo general, una valoración SPICE, de media, abarca 2 o 3 proyectos y aproximadamente entre 6 y 10 procesos.

De todas las valoraciones realizadas, los proyectos de Espacio han obtenido, por lo general, una valoración superior a proyectos de otros sectores. Los proyectos de Espacio se encuentran generalmente valorados en los niveles 3 y 4.

La capacidad de los procesos de ingeniería está muy equiparada entre proyectos de Espacio y otros sectores. Por el contrario, se han identificado las principales diferencias en los procesos correspondientes a las categorías de gestión (MAN), soporte (SUP), organización (ORG) y cliente (CUS).

Conclusiones

Un aspecto importante de la 15504 es que es una iniciativa internacional que se está desarrollando bajo el “paraguas” de la ISO. Actualmente, este estándar se encuentra en la transición hacia un estándar internacional que no sólo va a ser aplicable a los procesos software sino a cualquier tipo de procesos. De hecho, también ya se ha definido un modelo de referencia de procesos de acuerdo a la ISO 9001, denominado S9K.

La parte 5 de la ISO/IEC 15504 es un ejemplo de un modelo perfectamente aplicable en cualquier valoración de procesos software. Este modelo contiene una relación 1-a-1 con el modelo de referencia detallado en la 15504-2. Otros modelos de valoración, que cumplen con la 15504, es el SPICE for Space (S4S), que se está empleando ampliamente en la ESA y en todos sus proveedores. Actualmente, se está elaborando el Automotive SPICE, otro modelo de valoración aplicable para organizaciones del sector de la automoción.

Otro aspecto importante de las valoraciones 15504 es que el alcance se limita a unos proyectos determinados y a ciertos procesos dentro de los mismos. Además, los resultados de la valoración no se extrapolan al resto de la organización.

Aunque SPICE esté evolucionando, no hay que olvidar que su versión actual sigue siendo una extraordinaria herramienta para identificar mejoras de forma muy económica porque el alcance queda perfectamente identificado y delimitado antes de iniciar la valoración.

Bibliografía y referencias

[1] J. Müller, *SPICE Primer, versión 0.1, Draft*, 22.03.2004. www.synspace.com

[2] Synspace, *Synspace Training Course Notes*, March 2004.

[3] Synspace, *What every buyer should know – Capability Evaluation with SPICE for SPACE*, April 2002

[4] ISO, *ISO/IEC TR 15504-1:1998(E) Information Technology – Software process assessment – Part 1: Concepts and introductory guide*, 1998.

[5] ISO, *ISO/IEC TR 15504-2:1998(E) Information Technology – Software process assessment – Part 2: A reference model for processes and process capability*, 1998.

[6] ISO, *ISO/IEC TR 15504-3:1998(E) Information Technology – Software process assessment – Part 3: Performing an assessment*, 1998.

[7] ISO, *ISO/IEC TR 15504-4:1998(E) Information Technology – Software process assessment – Part 4: Guide to performing assessments*, 1998.

[8] ISO, *ISO/IEC TR 15504-5:1999(E) Information Technology – Software process assessment – Part 5: An assessment model and indicator guidance*, 1999.

[9] ISO, *ISO/IEC TR 15504-6:1998(E) Information Technology – Software process assessment – Part 6: Guide to competency of assessors*, 1998.

[10] ISO, *ISO/IEC TR 15504-7:1998(E) Information Technology – Software process assessment – Part 7: Guide for use in process improvement*, 1998.

[11] ISO, *ISO/IEC TR 15504-8:1998(E) Information Technology – Software process assessment – Part 8: Guide for use in determining supplier process capability*, 1998.

[12] ISO, *ISO/IEC TR 15504-9:1998(E) Information Technology – Software process assessment – Part 9: Vocabulary*, 1998.

[13] ISO, *ISO/IEC TR 15504-2:2003 Information Technology – Process assessment – Part 2: Performing an assessment*, 2003.

[14] ISO, *ISO/IEC TR 15504-3:2004 Information Technology – Process assessment – Part 3: Guidance on performing an assessment*, 2004.

[15] ISO, *ISO/IEC 12207:1995 Information Technology – Software life cycle processes –*. 1995

[16] ISO, *ISO/IEC 12207:1995/Amd1:2002*. 2002



personas



procesos



tecnologías

Cómo implantar CMMI y no morir en el intento

Cómo implantar CMMI y no morir en el intento

ARTÍCULO

Una guía rápida y fácil de entender, con las directrices necesarias para implantar CMMI... con éxito



Si hablamos de calidad en el ámbito de la tecnología de la información, en días de tendencias que hablan del afamado modelo CMMI (Capability Maturity Model Integrator). Este modelo americano lleva ya varios años en el mercado europeo y, por ende, en España, siendo ya un referente imprescindible, tanto, que comienza incluso a delimitar los clientes con los que una empresa puede trabajar y a ser una exigencia para poder prestar un determinado servicio.

Con este panorama, aquellas empresas desarrolladoras de software que quieren tener una presencia significativa en el mercado, se ven prácticamente abocadas a conseguir el preciado "galardón" del nivel de CMMI, de forma que les abra el acceso a algunos mercados y, además, sea una herramienta que les facilite el cumplimiento de sus objetivos de negocio. Pero conseguirlo no es fácil, y mucho más difícil es, aún, conseguir implantarlo con éxito, de forma que todos los procesos de la organización, métricas, herramientas, etc., que se definan, sean utilizados, cumplan su misión y faciliten la gestión de la compañía a la hora de dirigirse hacia el cumplimiento de sus objetivos.

Conocer la verdadera productividad del software desarrollado (los puntos función², casos de uso³ o número de líneas de código desarrollados por hombre y día), la estabilidad real del software entregado, y otros tantos indicadores que nos permitan predecir cómo serán los

función², casos de uso³ o número de líneas de código desarrollados por hombre y día, la estabilidad real del software entregado, y otros tantos indicadores que nos permitan predecir cómo serán los métodos de mantenimiento o garantía, son las cuestiones fundamentales que se abordan en la empresa, y se espera que la implantación de CMMI resuelva estas cuestiones?

Para que realmente CMMI sea respuesta a estas y otras cuestiones, es preciso tener en cuenta algunos aspectos que, sin duda, harán entender de forma clara en qué consiste el Modelo CMMI, y, sobre todo, cómo implantarlo con éxito.

Como cualquier instalación de un Sistema de Gestión, la implantación de CMMI requiere de la participación de recursos diversos (más personal, formación, herramientas, soporte externo, etc.), sobre todo, de una "impresión" de la dirección, fuerte y convencida en las bondades de la implantación del modelo CMMI. Si no está dispuesto a realizar la inversión, ¿qué mejor que hacerlo con garantías de conseguir el éxito y obtener un retorno económico significativo de la inversión realizada?

Autor: Anabel Manchón Díez Vocal del Comité de Software de la AEC Senior Manager en Mejora de Procesos. Sopra Profit S.A.U.

Publicado en la Revista CALIDAD N° 5, JULIO-AGOSTO 2008

Una guía rápida y fácil de entender, con las directrices necesarias para implantar CMMI... con éxito

Si hablamos de calidad en el ámbito de la Tecnologías de la Información, sin duda tendremos que hablar del afamado modelo CMMI (Capability Maturity Model Integrator)¹. Este modelo americano lleva ya varios años en el mercado europeo y, por ende, en España, siendo

ya un referente imprescindible, tanto, que comienza incluso a delimitar los clientes con los que una empresa puede trabajar y a ser una exigencia para poder prestar un determinado servicio.

Con este panorama, aquellas empresas desarrolladoras de software que quieren tener una presencia significativa en el mercado, se ven prácticamente abocadas a conseguir el preciado "galardón" del nivel de CMMI, de forma que les abra el acceso a algunos mercados y, además, sea una herramienta que les facilite el cumplimiento de sus objetivos de negocio. Pero conseguirlo no es fácil, y mucho más difícil es, aún, conseguir implantarlo con éxito, de forma que todos los procesos de la organización, métricas, herramientas, etc., que se definan, sean utilizados, cumplan su misión y faciliten la gestión de la compañía a la hora de dirigirse hacia el cumplimiento de sus objetivos.

Conocer la verdadera productividad del software desarrollado (los puntos función², casos de uso³ o número de líneas de código desarrollados por hombre y día), la estabilidad real del software entregado, y otros tantos indicadores que nos permitan predecir cómo serán los

¹ CMMI: Modelo de Buenas Prácticas para la Ingeniería de Software, sucesor de CMM (Capability Maturity Model) que fue desarrollado desde 1987 hasta 1997. En 2002 se crea la versión 1.1 de CMMI, y en Agosto de 2006 surge la v1.2 actualmente vigente a nivel mundial. CMMI surgió como fusión de diferentes modelos de buenas prácticas, para potenciar el uso y estandarización de los modelos de madurez. CMMI ha sido desarrollado por diferentes socios de la industria mundial, el gobierno americano y el SEI (Software Engineering Institute), además de contar con sponsors como la Secretaría de Defensa Americana y la Asociación Nacional Americana de Defensa.

² Punto Función: Medida estandarizada del tamaño de una aplicación software, basada en la funcionalidad del aplicativo solicitado por el cliente.

³ Caso de Uso: es una técnica para la captura de requisitos potenciales de un nuevo sistema o una actualización software. Cada caso de uso proporciona uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para conseguir un objetivo específico.

futuros desarrollos y estimar los periodos de mantenimiento o garantía, son las cuestiones fundamentales que se plantean en la empresa, y se espera que la implantación de CMMI resuelva. Pero, ¿realmente la implantación de CMMI resuelve estas cuestiones?



Para que realmente CMMI de respuesta a éstas y otras cuestiones, es preciso tener en cuenta algunos aspectos que, sin duda, harán entender de forma fácil en qué consiste el Modelo CMMI, y, sobre todo, cómo implantarlo con éxito.

Como cualquier instalación de un Sistema de Gestión, la implantación de CMMI requiere de la participación de recursos de diversa índole: personal, formación, herramientas, soporte experto, etc., y, sobre todo, de una “esponsorización” de la dirección, fuerte y convencida en las bondades de la implantación del modelo CMMI. Si se está dispuesto a realizar la inversión, ¿qué mejor que hacerlo con garantías de conseguir el éxito y obtener un retorno económico significativo de la inversión realizada?

El Modelo CMMI

CMMI (Capability Maturity Model Integration) es un modelo de referencia para la mejora de los procesos en compañías que desarrollan software, utiliza un lenguaje adaptado a los procesos específicos de la ingeniería y gestión software, y se estructura en 5 niveles de madurez. Bajo cada nivel de madurez se agrupan diferentes áreas de proceso, donde para cada proceso se definen los objetivos a satisfacer y se despliegan las prácticas a implementar

en la organización, por los diferentes proyectos y departamentos.

El Modelo CMMI presenta dos posibles representaciones: representación escalonada y representación continua. Veamos cada una de ellas:

Representación escalonada

Presenta 5 niveles de madurez, que se organizan de 1 a 5, de menor a mayor madurez:

- Las organizaciones del nivel 1 se caracterizan por un exceso de promesas, abandono de procesos en momentos de crisis y no ser capaces de repetir éxitos pasados.
- Las organizaciones de nivel 2 se caracterizan porque todos los proyectos de la organización han asegurado que los requisitos son gestionados y que los procesos se planifican, se ejecutan, miden y controlan.
- En el nivel 3 los procesos están bien caracterizados, comprendidos y se describen en estándares, procedimientos, herramientas y métodos. Los proyectos establecen sus procesos definidos, adaptando el conjunto de procesos estándares de la organización conforme a unas guías de adaptación.
- En el nivel 4 se establecen objetivos cuantitativos para calidad y ejecución de los procesos y se usan como criterios en la gestión de procesos y proyectos. Éstos se basan en las

necesidades de los clientes, usuarios finales, organización e implantadores de los procesos. La calidad y efectividad de los procesos se miden en términos estadísticos y se gestionan a través del ciclo de vida de los procesos.

- Por último, en el nivel 5 la organización mejora sus procesos sobre la base de una comprensión cuantitativa de las causas comunes de variación inherentes en los procesos y enfoca sobre el avance continuo de la eficacia del proceso, a través tanto de la mejora incremental como de la innovación tecnológica.

Representación continua

Los procesos se organizan en categorías de proceso, en vez de niveles de madurez, y se evalúa su nivel de capacidad. Presenta 6 niveles de capacidad, de 0 a 5, de menor a mayor:

- Nivel de Capacidad 0 o incompleto
- Nivel de Capacidad 1 o realizado
- Nivel de Capacidad 2 o gestionado
- Nivel de Capacidad 3 o definido
- Nivel de Capacidad 4 o gestionado cuantitativamente
- Nivel de Capacidad 5 u optimizando

Nivel de Madurez	Descripción	PROCESOS
1	Inicial	
2	Gestionado	<ul style="list-style-type: none"> • Gestión de requisitos • Planificación de proyectos • Control y seguimiento de proyectos • Gestión de acuerdos con proveedores • Gestión de la configuración • Aseguramiento de la calidad de procesos y productos • Métricas y análisis
3	Definido	<ul style="list-style-type: none"> • Desarrollo de requisitos • Solución técnica • Integración del producto • Verificación • Validación • Definición del proceso organizacional • Enfoque de la organización a procesos • Formación organizacional • Gestión integrada de proyectos • Gestión de riesgos • Análisis de decisiones y resolución
4	Gestionado cuantitativamente	<ul style="list-style-type: none"> • Gestión cuantitativa de proyectos • Rendimiento de los procesos de la organización
5	En optimización	<ul style="list-style-type: none"> • Análisis causal y resolución • Innovación organizacional y despliegue

A la hora de implantar CMMI es posible usar una u otra representación, si bien, la más extendida es la representación escalonada.

Por defecto, cualquier organización que nunca se ha evaluado frente al modelo CMMI estará en un nivel 1, para poder obtener el reconocimiento de un nivel de CMMI 2, 3, 4 ó 5, la organización ha de ser evaluada mediante un método específico homologado por el SEI⁴ (Software Engineering

Institute), denominado SCAMPI (Standard CMMI Appraisal Method Process Improvement) y es preciso contar con un equipo evaluador formado por personal de la empresa (conocedores

⁴ SEI: Software Engineering Institute es una organización financiada por el gobierno federal de USA, que centra sus investigaciones en las actividades de ingeniería de desarrollo software: adquisición, arquitectura de productos, mejora de procesos y medición de su desempeño, seguridad e interoperabilidad y la fiabilidad de los sistemas. Véase <http://www.sei.cmu.edu>

de sus procesos, de las operativas de los proyectos y del modelo CMMI) y por un Lead Appraiser (experto en el modelo CMMI y procesos de Gestión e Ingeniería de Software), capacitado por el SEI para poder dirigir los SCAMPI.

Se elija la representación que se elija, el modelo CMMI no nos dice CÓMO hay que hacer las cosas, pero sí nos dice QUÉ podemos hacer para mejorar.

Cómo implantar el modelo CMMI

Independientemente de la representación que se elija y del nivel de madurez al que se quiera optar, si se quiere implantar CMMI con éxito, no sólo hay que cumplir con los requisitos del modelo, sino que hay que tener en cuenta que los procesos a implantar, sean lógicos y realistas, tanto como para que se puedan cumplir en la organización. Hay que conseguir que el modelo funcione y retorne la inversión realizada, tanto económicamente como en ratios de calidad, para ello se han de tener en cuenta los siguientes aspectos:

1. Disponer en la organización de un “buen” sponsor

El principal aspecto a tener en cuenta en la implantación exitosa de CMMI es su “esponsorización”, el apoyo que desde la dirección de la empresa se ha de dar a esta iniciativa de mejora.

Todos los empleados de la organización deben conocer la iniciativa CMMI, ser conscientes de lo que implica (un cambio en sus actividades diarias, en sus herramientas, en su cultura) y asumir así desde la dirección los cambios a realizar.

Que nadie se plantee que se trata de una moda, el sponsor debe ser un referente en la empresa, alguien con el suficiente empuje y reconocimiento como para que nadie plantee la idoneidad de sus decisiones.

2. Objetivos definidos

El sponsor debe establecer y divulgar de forma clara los objetivos de mejora que se persiguen, no se trata de implantar CMMI como objetivo, sino que CMMI sea la herramienta que proporcione a la organización las vías para mejorar la productividad, la estabilidad del software liberado, etc.

3. Definir procesos con vocación de futuro

Tener en cuenta que no es la primera vez que una compañía desarrolla unos procesos perfectamente alineados con CMMI que finalmente no son llevados a la práctica nada más que durante los meses previos a la obtención del reconocimiento del Nivel de CMMI, pasándolos posteriormente al olvido, con la consiguiente pérdida económica y, cómo no, la infrutilización del modelo para poder obtener las métricas necesarias que orienten a la organización al cumplimiento de sus objetivos de negocio. Es en estos casos cuando desde la organización se critica el modelo y se dice que no funciona. Esta situación se debe evitar.

4. Asignar personal cualificado

Implantar CMMI no es una tarea fácil o intuitiva, la interpretación del modelo no es fácil y su comprensión no es sencilla. Hay que elegir a los mejores, a los más aptos y cualificados..., y, ¿por qué no?, confiar en los expertos externos.

En muchas ocasiones, el apoyo de este personal puede parecer caro pero, a la larga, con base

en los resultados que se obtienen, ahorra tiempo y dinero. ¿Por qué investigar hacia dónde ir, cómo llegar...? En definitiva, ¿por qué empeñarse en inventar la rueda cuando alguien ya la inventó? ¡Hay que aprobar la experiencia de otros para conseguir el éxito!

5. Impulsar la gestión del cambio

Como último aspecto importante a destacar, la organización debe gestionar la aversión inicial al cambio en las nuevas operativas que se definirán, ha de facilitar y supervisar la adherencia a los procesos establecidos e involucrar a todos los afectados en la definición y ejecución de tareas, de forma que toda la organización se sienta partícipe del cambio y lo haga propio, sin verlo como una imposición.

Si conseguimos tener en cuenta estos aspectos y sortear las vicisitudes a las que sin duda nos tendremos que enfrentar, en un periodo de tiempo más o menos largo, es factible obtener el preciado nivel de CMMI. El SEI orienta a las organizaciones en lo referente al tiempo estimado para alcanzar los diferentes niveles de madurez:

- 12 meses para pasar de nivel 1 a 2.
- 18 meses para alcanzar el nivel 3.

Es importante destacar que la consecución del nivel de CMMI no es el final del Proyecto de Mejora. Si se desea una implantación exitosa, no se debe considerar como un punto final, sino como el primero de los pasos a seguir en la mejora continua.

El ir aumentando en el nivel de CMMI, mejorando la madurez de la organización, repercutirá en mejores prácticas en la organización, en métricas más sólidas que proporcionarán mayor información para gestionar los procesos y proyectos, en obtener mejores resultados en los productos y, cómo no, en mejorar resultados económicos y cuota de mercado.



personas



procesos



tecnologías

CMMI DEV Diferencias entre nivel 2 y nivel 3 y una estrategia de implantación



Cecilia Rigoni
Gerente de Caelum,
Information & Quality
Technologies.
Vocal del Comité CSTIC
de la AEC

El modelo CMMI DEV, cuyo objetivo es servir de guía para mejorar el proceso de desarrollo de software, puede aplicarse de dos formas distintas: una utilizándolo para mejorar algunas actividades, cuyo conjunto corresponde a una de las llamadas áreas de proceso, hasta alcanzar un nivel esperado y otra mejorando un grupo establecido de actividades, organizadas en áreas de proceso. Las distintas formas de aplicar el modelo se denominan representaciones. El primero es la "representación continua" y el segundo, la "representación por etapas".

En la representación por etapas, la preferida hasta ahora por las organizaciones, se primer nivel que se evalúa es el 2 y se llega hasta el 5. Existe cierta desorientación sobre las exigencias del modelo en los niveles 2 y 3, que este artículo pretende aclarar.

En el nivel 2 hay siete áreas de proceso:

- Gestión de Requerimientos (REQM, Ingeniería).
- Planificación de Proyecto (PP, Gestión de proyectos).
- Monitorización y Control de Proyecto (PMC, Gestión de proyectos).
- Gestión de Acuerdos con Proveedores (SAM, Gestión de proyectos).
- Medición y Análisis (MA, Soporte).
- Aseguramiento de la Calidad de Proceso y de Producto (PPQA, Soporte).
- Gestión de Configuración (CM, Soporte).

En el nivel 3 a esas siete áreas se le agregan otras once:

- Gestión Integrada de Proyecto + IPPD (IPM + IPPD Gestión de proyectos).
- Definición de Procesos de la Organización + IPPD (OPD + IPPD Gestión de Proyectos).
- Enfoque en los Procesos Organizativos (OPF Gestión de proyectos).
- Gestión de Riesgos (RSKM, Gestión de proyectos).
- Formación Organizativa (OT, Gestión de proyectos).
- Integración de Producto (PI, Ingeniería).
- Desarrollo de Requerimientos (RD, Ingeniería).
- Solución Técnica (TS, Ingeniería).
- Validación (VAL, Ingeniería).
- Verificación (VER, Ingeniería).
- Análisis de Decisiones y Resolución (DAR, Soporte).

octubre-diciembre 2010 CALIDAD 59

Autor: Cecilia Rigoni Gerente de Caelum, Information & Quality Technologies.

Publicado en la Revista Calidad - n° IV • 2010

El modelo CMMI DEV, cuyo objetivo es servir de guía para mejorar el proceso de desarrollo de software, puede aplicarse de dos formas distintas: una utilizándolo para mejorar algunas actividades, cuyo conjunto corresponde a una de las llamadas áreas de proceso, hasta alcanzar un nivel esperado y otra mejorando un grupo establecido de actividades, organizadas en áreas de proceso. Los distintos caminos se denominan representaciones. El primero es la "representación continua" y el segundo, la "representación por etapas".

En la representación por etapas, la preferida hasta ahora por las organizaciones, el primer nivel

que se evalúa es el 2 y se llega hasta el 5. Existe cierta desorientación sobre las exigencias del modelo en los niveles 2 y 3, que este artículo pretende aclarar.

En el nivel 2 hay siete áreas de proceso:

- Gestión de Requerimientos (REQM, Ingeniería).
- Planificación de Proyecto (PP, Gestión de proyectos).
- Monitorización y Control de Proyecto (PMC, Gestión de proyectos).
- Gestión de Acuerdos con Proveedores (SAM, Gestión de proyectos).
- Medición y Análisis (MA, Soporte).
- Aseguramiento de la Calidad de Proceso y de Producto (PPQA, Soporte).
- Gestión de Configuración (CM, Soporte).

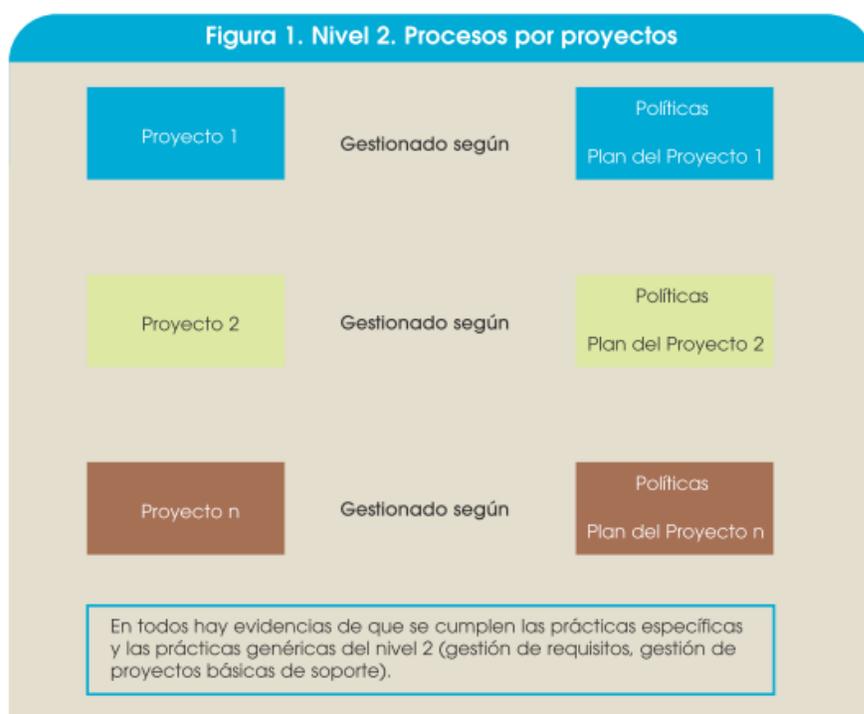
En el nivel 3 a esas siete áreas se le agregan otras once:

- Gestión Integrada de Proyecto + IPPD (IPM + IPPD Gestión de proyectos).
- Definición de Procesos de la Organización + IPPD (OPD + IPPD Gestión de Proyectos).
- Enfoque en los Procesos Organizativos (OPF Gestión de proyectos).
- Gestión de Riesgos (RSKM, Gestión de proyectos).
- Formación Organizativa (OT, Gestión de proyectos).
- Integración de Producto (PI, Ingeniería).
- Desarrollo de Requerimientos (RD, Ingeniería).
- Solución Técnica (TS, Ingeniería).
- Validación (VAL, Ingeniería).
- Verificación (VER, Ingeniería).
- Análisis de Decisiones y Resolución (DAR, Soporte).

Además de la diferencia entre áreas de proceso, hay entre estos dos niveles una diferencia conceptual importante que tiene que ver con la forma en que están implementados los procesos. La diferencia más importante es el ámbito en que el proceso de desarrollo de software está implantado. Para aclarar esta cuestión, es conveniente ver qué es lo que se exige en cada nivel.

Nivel 2

En el nivel 2 se pide que los proyectos se gestionen. Esto quiere decir que el equipo de proyecto debe trabajar siguiendo un plan documentado y unas políticas emergidas de la dirección de la organización, donde se especifiquen las directrices para desarrollar software. Estas políticas no deben confundirse con procesos, son indicaciones generales que deben seguirse sin discusión, son las leyes. También las prácticas genéricas tienen que estar implantadas en el proyecto (figura 1).



Ahora bien, ¿dónde se especifican los procesos? ¿Cómo puede conocerlos un participante en el equipo de desarrollo? El documento fundamental es el plan de proyecto donde se reúnen todos los planes necesarios para el proyecto.

El contenido del plan del proyecto más un conjunto de actuaciones de formación y seguimiento de los

propios procesos serán los que darán forma a la gestión del proyecto y al desarrollo de los productos. Esto último indica que hay dos grupos de actividades que se tienen que planificar: una es la de desarrollo del producto, que normalmente se hace siguiendo una metodología, y otra es la de gestión del proyecto, que es la que entre otras cosas determina la metodología a seguir en el desarrollo del producto⁵.

Para conseguir un nivel 2 se tienen que evidenciar que los requisitos y los proyectos se

⁵ En organizaciones poco maduras, es bastante normal que se entienda como plan de proyecto lo que solamente es el plan de desarrollo del producto y que no se defina la forma en que se va a gestionar el proyecto. Ambos planes tienen que hacerse simultáneamente y la coordinación de las actividades de gestión y las de desarrollo es lo que asegura el cumplimiento de plazos y costes. Un ejemplo: Para el desarrollo del producto es necesaria la utilización de una herramienta. En el plan de gestión tienen que estar establecido el mecanismo y los responsables de la compra, que son los que asegurarán que la herramienta esté operativa en el momento en que se necesite. Si la herramienta no está disponible en la fecha prevista, se retrasará la construcción del producto y, por tanto, no se cumplirá el plan de desarrollo del mismo.

gestionan. Naturalmente, se tienen que hacer las actividades de ingeniería, si no, no habría producto, pero en este nivel esas tareas no son evaluadas.

Un ejemplo de que el modelo se ha hecho como resultado de la observación de la realidad de las organizaciones es haber notado que es fundamental conseguir que los procesos de ingeniería, generalmente definidos y adecuados, no se vean desvirtuados por una mala gestión y por eso en el primer nivel de madurez se asegura de que esta carencia no está presente.

El modelo pide que todos los proyectos se gestionen, pero no que todos se gestionen de la misma forma. Cada proyecto debe tener su plan en el que se especifica de qué manera se gestionarán los requisitos, los datos y la configuración, cómo se reciben, aprueban e incorporan los cambios, cómo se hace el seguimiento del plan de proyecto, qué medidas se toman y cómo se asegura la calidad del producto y de los procesos y, si se tienen proveedores, cómo se asegura y controla el cumplimiento del contrato.

Además se tendrán que implantar en cada proyecto las prácticas genéricas.

Cada proyecto tendrá que:

- tener sus políticas;
- haber planificado las actividades de cada una de las áreas de proceso (responsables, esfuerzo y duración durante todo el proyecto);
- tener recursos humanos y materiales (personas, herramientas) apropiados;
- haber formado a las personas para que puedan hacer los procesos como se ha previsto en el plan;
- haber establecido cómo se van a controlar los documentos que se producen en cada proceso;
- haber asignado responsables para cada proceso;
- asegurar la participación de los equipos externos al proyecto que tengan que intervenir;
- asegurar con objetividad que los procesos se llevan a cabo tal y como se estipuló en el plan de proyecto (revisan personas distintas de las que realizan el proceso);
- haber monitorizado y controlado los procesos día a día;
- haber revisado con el nivel directivo las actividades, el estado y los resultados del proceso.

Para un proceso gestionado, las descripciones del proceso, los estándares y los procedimientos son aplicables a un proyecto particular, grupo o función de la organización. Como resultado, dos proyectos en una organización pueden ser gestionados de diferentes formas.

Nivel 3

Si se quiere alcanzar el nivel 3 de madurez, el modelo pide que los proyectos sigan un proceso definido.

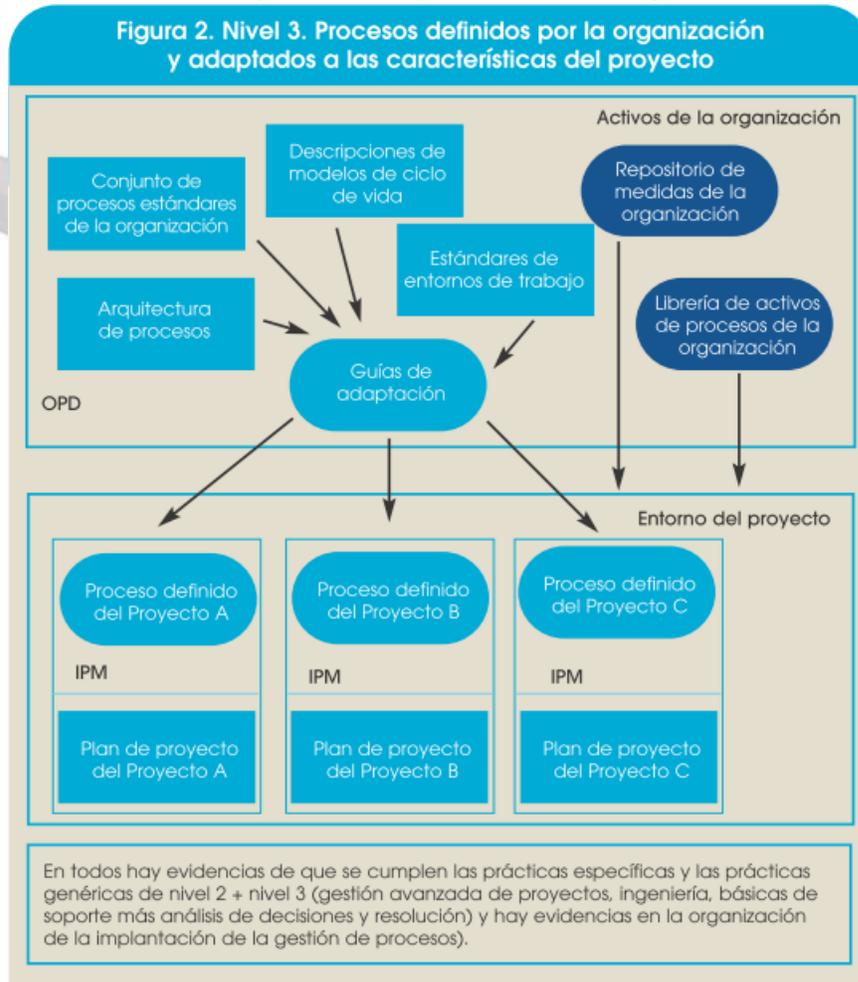
Para ello, la organización tiene que haber establecido un conjunto de procesos estándares y también unas guías para adaptar esos procesos estándares a las particularidades de los proyectos. Cada proyecto selecciona los procesos estándares siguiendo las guías y define su propio proceso.

En este caso, tanto los documentos como las actividades estarán especificados en los procesos estándares, y no habrá necesidad de “inventar” los procesos de gestión y de desarrollo cuando se comienza el proyecto (figura 2).

El nivel 3 incorpora todas las actividades de ingeniería (desarrollo de requisitos, solución técnica, integración del producto, validación y verificación, etc.) más las de gestión de

requisitos, gestión del proyecto y soporte que se vieron en el nivel 2.

El modelo, además de pedir que los proyectos se gestionen siguiendo un proceso definido a partir de los procesos estándares, pide que se hayan establecido una serie de actividades destinadas a la mejora de los procesos de la organización. Las lecciones aprendidas, las



medidas de rendimiento de los procesos, las sugerencias de las personas que utilizan los procesos, son elementos que van a servir para ir mejorándolos.

Las actividades relacionadas con la definición de los procesos tienen que estar establecidas formalmente para organizar, planificar, implantar y desplegar las mejoras, definir los procesos y preparar e impartir formación sobre los mismos.

Por último, en este nivel se exige que las decisiones críticas se tomen siguiendo un proceso que minimice la subjetividad. Se

deberán especificar los criterios para conocer ante qué tipo de decisiones se debe seguir el proceso establecido y mantener una guía a seguir para el análisis de las alternativas y razones que lleven a tomar una decisión.

En cuanto a las prácticas genéricas, en este nivel se mantienen las diez exigidas para alcanzar el nivel 2 y se incorporan las que dan la clave de este nivel:

- Establecer y mantener la descripción de un proceso definido.
- Recoger productos de trabajo, medidas, resultados de medición e información de mejora procedente de la planificación y realización del proceso para dar soporte al uso futuro y a la mejora de los procesos y de los activos de proceso de la organización.

De nivel 2 a nivel 3

Normalmente, en las organizaciones que quieren evaluarse para conocer su grado de madurez existen ya una serie de procesos establecidos, aunque de manera desordenada e implantados parcialmente.

Cuando se empieza el proyecto de mejora para alcanzar el nivel 2 se suelen definir procesos

estándares para toda la organización. Esa definición no la pide el modelo, se hace para poder conseguir desarrollos disciplinados que no dependan de las habilidades del jefe de proyecto. De esta manera, se están generando elementos que son de gestión de procesos, y que se piden para el nivel 3 de madurez. Igualmente la formación suele ser organizativa.

Es interesante entender las áreas básicas de gestión de proceso cuando se inicia la mejora, aunque ésta sea para alcanzar el nivel 2. Las prácticas de las áreas Enfoque en los Procesos Organizativos, Definición de los Procesos Organizativos y Formación Organizativa son las que se tienen que aplicar para conseguir un conjunto adecuado de procesos y, si se respetan, se podrán presentar evidencias de su cumplimiento en una evaluación de nivel 3 de madurez.

En el paso de nivel 2 a 3 se tendrán, por tanto, los procesos de gestión de requisitos, los básicos de gestión de proyectos y de soporte ya definidos e implantados. Habrá que definir los procesos de ingeniería, los avanzados de gestión de proyectos (gestión integrada de proyecto, gestión de riesgos) y en soporte, análisis de decisiones y resolución.

La diferencia entre el nivel 2 y el 3 puede parecer fácilmente salvable, pero no lo es tanto. Los procesos de nivel 2 tienen que estar perfectamente institucionalizados y se tienen que institucionalizar también todos los de nivel 3, incluyendo aquellos que son de gestión de procesos y de toma de decisiones. Se tienen que tener evidencias de que se mantiene la base de activos de procesos de la organización y de que se estudian e incorporan mejoras en los procesos.

Quizás el área más difícil de entender sea Análisis de Decisiones y Resolución, aunque su objetivo es muy importante.

Se trata de eliminar la subjetividad de las decisiones y de tener constancia de las razones que llevaron a tomarlas. Cualquier persona cuyas responsabilidades incluyan la toma de decisiones críticas, debería agradecer la inclusión de las prácticas de esta área. En organizaciones medianas y pequeñas no siempre se ve su utilidad y tampoco es fácil conseguir las evidencias.

Las prácticas genéricas 2.1 a 2.10 tienen que cumplirse para todas las áreas específicas de nivel 2 y 3, y además se tienen que evidenciar las prácticas genéricas 3.1 y 3.2. La primera consiste en tener un proceso definido para cada área de proceso y la segunda, en recopilar productos de trabajo, medidas, e información para mejorar los procesos y los activos de proceso.

Estrategia de implantación

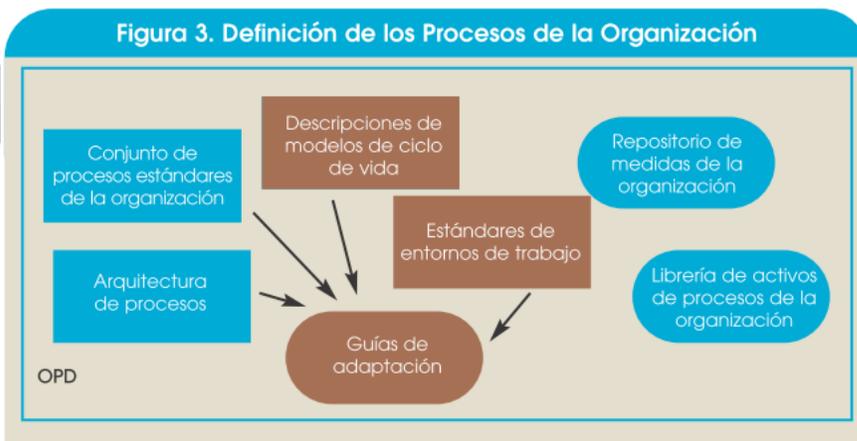
En el momento en que se empieza el proyecto de mejora con el objetivo de alcanzar el nivel 2 de madurez es importante empezar a preparar la implantación de las áreas orientadas a los procesos organizativos Enfoque en los Procesos Organizativos, Definición de los Procesos Organizativos y Formación Organizativa (OPF, OPD y OT), siguiendo las prácticas específicas de las mismas. De esta forma, se tendrán definidos los procesos de nivel 2 en el nivel organizativo y se estarán generando evidencias de nivel 3 para la áreas mencionadas.

Una vez alcanzado el objetivo de nivel 2 se tiene que definir el resto de procesos de ingeniería y soporte que se evalúan para nivel 3.

Es importante que en los dos periodos se creen evidencias que demuestren que se están monitorizando, analizando y mejorando los procesos a partir de los datos obtenidos durante el desarrollo de los proyectos para evidenciar que se hacen las prácticas genéricas de nivel 3.

Para nivel 2

- Planificar el proyecto de mejora siguiendo las prácticas específicas del área Enfoque en los Procesos

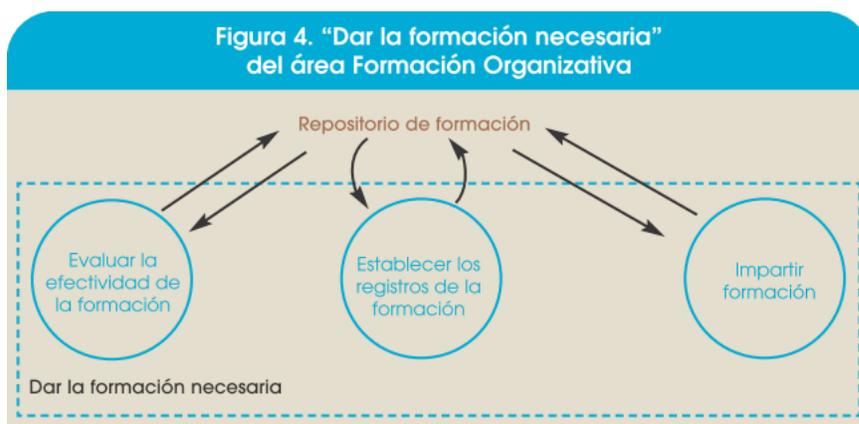


Organizativos para las áreas de nivel 2.

- Definir los procesos de la organización que incluyen las prácticas de nivel 2, intentando seguir las prácticas específicas del área Definición de los Procesos de la Organización que

están en azul en la figura 3.

- Formar en los procesos definidos, intentando seguir las prácticas específicas para la meta



"Dar la formación necesaria" del área Formación Organizativa (figura 4).

- Guardar las evidencias de las prácticas de las áreas de Gestión de Procesos.

De esta forma se va haciendo una primera

aproximación a las prácticas de gestión de procesos que se completarán para nivel 3.

Para nivel 3

- Redefinir los procesos de gestión de procesos asegurando que se cumplen las metas de OPE, OPD y OT.
- Definir los procesos que incluyen las prácticas de nivel 3.
- Gestionar los proyectos adaptando los procesos estándares según lo establecen las guía.
- Implantar las prácticas genéricas 3.1 y 3.2.

Riesgos y Auditorías

La Auditoría de calidad es un examen metódico, independiente y planificado de los elementos de un proceso, producto, procedimiento u organización para determinar si las actividades y resultados relativos a la calidad satisfacen las disposiciones previamente establecidas, y si estas disposiciones se llevan a cabo efectivamente y son aptas para alcanzar los objetivos.

Es importante destacar que:

- Examen metódico, implica el uso de un método definido, consistente y estable a lo largo del tiempo.
- Examen independiente, debe ser realizado por una persona o equipo independiente y por tanto sin relación jerárquica o funcional con el auditado.
- Examen planificado, las auditorías tienen un carácter preventivo, es decir, la Auditoría no se realiza como consecuencia de la aparición de un problema, sino que tiene una periodicidad, que depende del tipo de Auditoría a realizar. En la Auditoría de Calidad el auditado conoce con antelación las fechas en que será auditado, y para ello, el Departamento de Calidad deberá editar a primeros de año la lista de los procedimientos y procesos que serán auditados ese año y en que fechas.

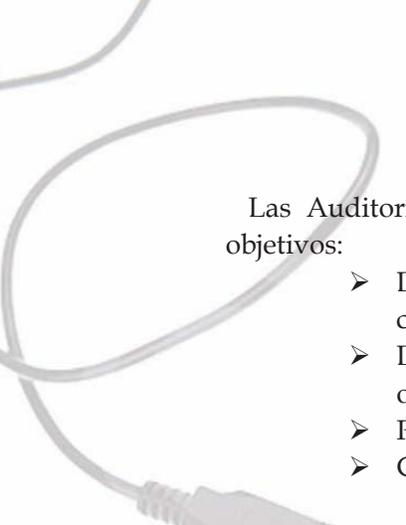
La Auditoría debe aplicarse, siempre que estén procedimentados, regulados o documentados, a las políticas, prácticas o modos de operación, productos y servicios que se prestan y no sólo a los procedimientos.

La Auditoría deberá llevarse a cabo por personal que no tenga relación con el departamento a auditar. Las Auditorías de calidad pueden programarse y realizarse a todo el sistema en su globalidad o a una o varias partes del sistema. El que sea global o parcial dependerá de los problemas existentes, de la importancia de la actividad o actividades sometidas a Auditoría y de los requisitos del sistema.

La frecuencia de las auditorías internas puede variar de un procedimiento a otro en función de:

- La importancia del procedimiento o del proceso,
- Las posibilidades de variación,
- Los resultados de las auditorías precedentes,
- La disponibilidad de recursos,
- La criticidad de los proyectos.

Los resultados de las Auditorías deben registrarse y transmitirse al personal responsable del departamento auditado. Las acciones correctoras derivadas de la Auditoría deben realizarse cuanto antes, corrigiendo las deficiencias encontradas.



Las Auditorías se realizan generalmente para conseguir uno o varios de los siguientes objetivos:

- Determinar la conformidad o no conformidad de los elementos del sistema de la calidad con los requisitos especificados.
- Determinar la eficacia del sistema de la calidad implantado para alcanzar los objetivos de la calidad especificados.
- Proporcionar al auditado la oportunidad de mejorar su sistema de la calidad.
- Cumplir los requisitos

Estas Auditorías pueden iniciarse con regularidad o como consecuencia de cambios importantes en el sistema de la calidad del organismo o de la calidad de sus procesos, productos o servicios o para el seguimiento de las acciones correctivas.

Las auditorías internas siempre son realizadas por miembros del Departamento de Calidad.

Debe existir un sistema o método para desarrollar las Auditorías.

La Auditoría es confidencial y el auditor o equipo auditor, informará de los requisitos, documentará las observaciones y elaborará un informe de resultado de la Auditoría.

En la Auditoría es muy importante entablar una relación de igualdad, no se trata de una gestión “policial”, es un proceso técnico, en el cual el auditado tiene que ver su beneficio al recibir la información sobre la marcha de su trabajo.

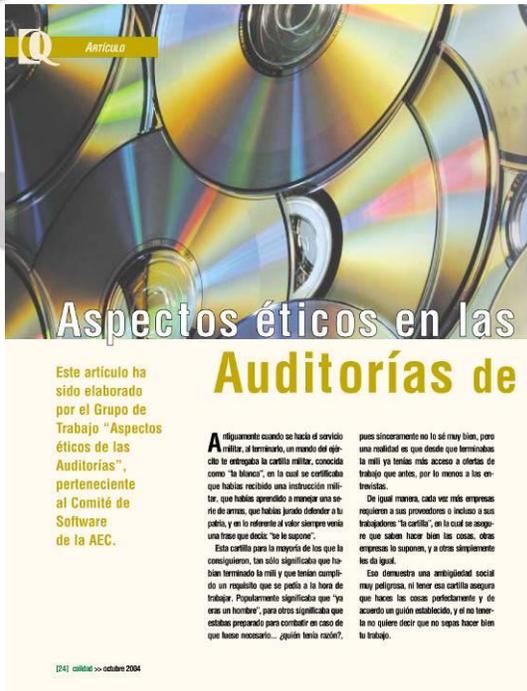
La calidad del software lleva pareja la actividad de la auditoria; un profesional norteamericano definió un sistema de calidad de la siguiente manera: “Primero escribe que quieres hacer”, segundo “haz lo que has escrito” y Tercero y más importante, alguien (un tercero) que no te conozca que compruebe que efectivamente lo estás haciendo”

Artículos presentados en esta selección:

- [Aspectos éticos en las Auditorías de Calidad de Software](#)
- [El derecho a creer Gestión de Riesgos en Proyectos Software El derecho a creer](#)
- [Gestión de la incertidumbre en el proceso de estimación durante el ciclo de vida](#)
- [Esta web ha sido atacada](#)

Aspectos éticos en las Auditorías de Calidad de Software

Autor: Este artículo ha sido elaborado por el Grupo de Trabajo “Aspectos éticos de las Auditorías”, perteneciente al Comité de Software de la AEC.



Publicado en la Revista Calidad - Octubre 2004

Antiguamente cuando se hacía el servicio militar, al terminarlo, un mando del ejército te entregaba la cartilla militar, conocida como “la blanca”, en la cual se certificaba que habías recibido una instrucción militar, que habías aprendido a manejar una serie de armas, que habías jurado defender a tu patria, y en lo referente al valor siempre venía una frase que decía: “se le supone”.

Esta cartilla para la mayoría de los que la consiguieron, tan sólo significaba que habían terminado la mili y que tenían cumplido un requisito que se pedía a la hora de trabajar.

Popularmente significaba que “ya eras un hombre”, para otros significaba que estabas preparado para combatir en caso de que fuese necesario... ¿quién tenía razón?, pues sinceramente no lo sé muy bien, pero una realidad es que desde que terminabas la mili ya tenías más acceso a ofertas de trabajo que antes, por lo menos a las entrevistas.

De igual manera, cada vez más empresas requieren a sus proveedores o incluso a sus trabajadores “la cartilla”, en la cual se asegure que saben hacer bien las cosas, otras empresas lo suponen, y a otras simplemente les da igual.

Eso demuestra una ambigüedad social muy peligrosa, ni tener esa cartilla asegura que haces las cosas perfectamente y de acuerdo un guión establecido, y el no tenerla no quiere decir que no sepas hacer bien tu trabajo.

Para ello se requiere siempre a un tercero, que es el que definitivamente decide en qué manera se cumple una normativa o no.

Si a esta “metáfora” le ponemos nombre, se ve claramente que la cartilla puede ser perfectamente un sistema de calidad basado en una norma ISO, generalmente la 9001:2000; y que ese “tercero” es una entidad certificadora.

Por tanto, es correcto afirmar que Certificar un sistema de calidad consiste básicamente en que una entidad acreditada, competente, afirma que una empresa hace las cosas de acuerdo a una norma y da fe de ello por escrito, sabiendo que todo lo que certifica, se tiene que revisar y volver a certificar a medio o largo plazo.

A todas estas definiciones, hay que añadirle un componente más: las características esenciales de los productos que se hacen en nuestro sector, el software, es decir:

- se hacen productos “invisibles”,
- no se realizan mediante un método universal y

- además son muy difíciles de probar en su totalidad, debido al constante cambio al que están sometidos.

Para obtener un certificado o acreditación hay que seguir una serie de pasos llevados a cabo por diferentes actores. Existe un primer nivel de entidades que son aquellas que acreditan a otras entidades para que puedan certificar. En España esta entidad es ENAC.

Las entidades de certificación son aquellas que emiten certificados para las empresas después de haber realizado las oportunas auditorías. En España hay en total catorce entidades acreditadas en el sector de las actividades informáticas.

Estas entidades escogen o subcontratan auditores que evalúan a las empresas en relación a unas especificaciones o normas como pueden ser la ISO 9001:2000, la ISO 14000, etc.

Las auditorías de calidad tienen en la actualidad un gran protagonismo, motivado por el impulso que la certificación ha adquirido en los últimos años. Ello ha conducido a grandes organizaciones como ISO a que se desarrollen normas o métodos para llevar a cabo auditorías tanto internas como externas, como son las normas ISO 19011, que demarcan una serie de directrices para la realización de auditorías, proporcionando orientación para la gestión de las auditorías, la realización de las mismas así como de la competencia y la evaluación de los auditores.

A la vista de este escenario, aparentemente bien pensado, y a razón de la situación actual de las empresas de software surge una pregunta: ¿qué credibilidad nos ofrece el certificado recibido?

Evidentemente el certificado es un documento firmado, por lo tanto la credibilidad del certificado es la confianza que nos merezca el auditor y la certificadora. Y ¿cómo se consigue esa confianza?, bueno para contestar a esta pregunta hay que conocer el objetivo que se persigue con dicho certificado.

Inicialmente las empresas piensan que hacen las cosas “muy bien”, los clientes quieren garantías de que su trabajo “esté bien hecho” y “a tiempo”, los inversores quieren conseguir “más dinero”, los proveedores negocio y continuidad y los trabajadores un trabajo digno.

Por lo tanto, este certificado nunca debe ser un mero trámite, ya que en caso contrario “la estafa” sería de unas dimensiones considerables, y el número de afectados incalculables.

Por ello es muy importante que la auditoría no sea frívola o incompleta, y que el auditor tenga el debido cuidado profesional así como una conducta ética no actuando nunca de manera incompetente o desconociendo el sistema que debe auditar.

Para ello debe de tener muy claro además del método a utilizar, el alcance, los objetivos y los criterios de auditoría con los que va a llevarla a cabo.

En España existen multitud de títulos tanto a nivel personal como institucional de auditores, pero a diferencia de otros sectores muy similares como el de la seguridad donde hay títulos específicos para los auditores como el CISA o CIA, por ejemplo, para auditar empresas de software no hay ninguna acreditación específica.

De ahí el creciente interés por modelos como el CMM o ITIL que centran su validez y credibilidad en la validación de personas con cualificación informática, respaldados por organismos informáticos como el SEI y con métodos de evaluación universales y reconocidos mundialmente como SCAMPI.

No obstante, el hacer las cosas bien implica dinero, tanto a nivel de formación, como en dedicación a montar sistemas serios, y parece ser que en España (analizando los datos de encuestas realizadas en el sector) las empresas que desarrollan sistemas dan prioridad al hardware o a la parte no software, mostrando, en muchos de los casos, algún interés por el

“sello”, que casi siempre debe presentar en alguna “suculenta” oferta.

Esto no quiere decir que todos los certificados emitidos en empresas certificadoras de software sean banales, pero es evidente que este tema debe cambiar.

El desconocimiento mostrado por excelentes auditores en otras disciplinas de calidad, en el mundo informático, la falta de conocimiento público de los códigos éticos de las certificadoras y el interés de marketing a veces excesivo por parte de las empresas por disponer “del sello”, al único sitio que nos lleva es a un desprestigio no sólo de los sistemas de calidad de software (principalmente las normas ISO), sino de los profesionales que trabajamos en calidad dentro de este ajetreado sector.

Un ingeniero de calidad es tan importante para un proyecto informático como el mejor de los programadores, a partir de que las direcciones asuman este rol, no solamente las empresas saldrán ganando sino que el proceso de auditorías y sus resultados, es decir, los certificados, ganarán en respeto y dignidad dentro de la sociedad de la Información.

Quizá para ello será necesario que algunas direcciones cambien su concepción sobre la calidad del software, considerándola como el objetivo de sus esfuerzos, pero antes, todo ello pasa porque las certificadoras concedan sólo los certificados a las empresas que realmente estén preparadas para obtenerlos, quizá siguiendo un símil de evaluación que tenga en cuenta no sólo las evidencias objetivas que se pueden generar con relativa facilidad, sino las percepciones que expertos auditores en calidad del software obtengan de entrevistas al personal de todos los niveles de la organización.

Deben llegar a comprender que el certificado debe ser una consecuencia y nunca un objetivo, la imagen actual que los productos software ofrecen a los usuarios nos debe hacer reflexionar sobre si realmente nos han auditado o nos han mirado los documentos y las acciones correctivas nada más, sin pararse a verificar si hay diseño, revisiones, pruebas..., en fin todo aquello que en otros sectores es obvio y que desde luego a la hora de auditarse se mira muy a fondo.

La fiabilidad del proceso de auditorías de calidad de software y la confianza en el mismo dependerá siempre de la competencia de aquéllos que llevan a cabo la auditoría, y esa competencia debe basarse no sólo en las cualidades personales, sino en la aptitud para aplicar los conocimientos adquiridos mediante la formación y la experiencia laboral en el sector adecuado, la formación como auditor y la experiencia en auditorías.

Quizá una solución para disponer de auditores de software más capacitados fuese la exigencia de conocimientos del software específicos para este tipo de auditores, además de disponer de la experiencia demostrable y adecuada para auditar este tipo de negocio avaladas por algún organismo supracertificador.



personas



procesos



tecnologías

El derecho a creer Gestión de Riesgos en Proyectos Software

El derecho a creer



Autor: JOSÉ BARATO / Principal Consultant, Atos Consulting. Vocal del Comité de Software de la AEC

Publicado en la Revista CALIDAD Nº 4, JUNIO 2006

“Un barco de emigrantes está a punto de hacerse a la mar con un cargamento completo de pasajeros. El armador está tan preocupado por el estado de su viejo barco (no muy bien construido en origen), que se cuestiona si podrá soportar un último pasaje. Con poco esfuerzo, sin embargo, supera sus dudas y se convence a sí mismo de que no pasa nada por un pasaje más. El barco, después de todo, ya soportó en su día algunas tempestades y siempre se las arregló para llegar a puerto. ¿Por qué no una vez más? El barco sale a la mar, se

hunde y mueren todos. ¿Qué puede decirse del armador? Seguramente esto: que es culpable de la muerte de aquellos hombres. Si bien se admite que su convencimiento era honesto y verdadero, esto no puede ayudarle porque no tenía derecho a creer sobre la base de tan pobres evidencias. Su creencia fue adquirida no a partir de una paciente investigación, sino ahogando sus dudas. Y aunque al final él se sentía tan seguro que no podía pensar de otra manera, dado que él mismo se había esforzado conscientemente para adquirir ese estado mental, debe ser culpado como responsable.

Supongan ahora que el barco se las arregla después de todo para completar el viaje sin pérdidas de vidas humanas.

¿Es menos culpable el armador? Ni un ápice. Cuando algo se hace una vez, está bien o mal para siempre, independientemente de las consecuencias. El armador no habría sido inocente, simplemente no le habrían descubierto.

Lo que se decide creer no debe estar exento del juicio ético de los demás: La mera creencia es suficiente para que uno cargue con las consecuencias del comportamiento no ético, si no tiene ‘el derecho a creer’ lo que decide creer.”

¿Qué es Gestión de Riesgos?

El texto anterior pertenece al ensayo The Ethics of Belief, de William Clifford, 1877. Antes de Clifford, se aceptaba que las creencias no podían ser consideradas a la luz del juicio ético. Se podía creer cualquier cosa que uno quisiera. Uno podía incluso creer que era posible hacer seis cosas imposibles antes del desayuno, como hacía la Reina Blanca en el libro de Lewis Carroll A

través del espejo.

Un caso parecido al del barco de emigrantes podría ser el siguiente:

Tu jefe te pide que consideres hacerte cargo de un proyecto que tiene que terminar en Navidad, con un equipo de sólo 3 personas. Tú expresas tus dudas sobre tan ajustado plazo. Tu jefe responde: "Por eso he pensado en ti". Tú probablemente aceptarás el proyecto, la oportunidad, el prestigio... Pero también tendrás que creerte la planificación, ése es el precio. Más tarde, darás rienda suelta a tus creencias. ¿Por qué no en Navidad? Otros proyectos han terminado en plazos reducidos, ¿no? Antes de que te des cuenta, puede que te sientas confiado. El tiempo probará lo contrario, sin duda, pero de momento, estás prácticamente seguro de que se puede realizar el trabajo. Sí, esto es lo que crees, ¿pero tienes derecho a creerlo?

Probablemente no hay trabajo en el mundo para el que sea más necesario tener que creer en seis cosas imposibles antes del desayuno, que la gestión de proyectos software. Día tras día, nos convencemos a nosotros mismos para creer en un plazo, un presupuesto o un factor de productividad, que más tarde el tiempo demuestra que eran completamente imposibles.

La ciencia que se ocupa de creer sólo lo que se tiene derecho a creer se llama Gestión de Riesgos

Esta disciplina esencial aplica los principios éticos sobre las creencias de Clifford a cualquier esfuerzo que contenga elementos de incertidumbre. A Vd. le servirá como guía a través de dicho esfuerzo (un proyecto de software, por ejemplo) para eliminar el entramado de pequeñas mentiras y auto-embaucamientos que han trabado su trabajo en el pasado. Será su alternativa para no creer "seis cosas imposibles antes del desayuno".

Gestión de Riesgos es Gestión de Proyectos para adultos

Una característica que define la madurez de las personas es nuestra facultad para hacerle frente a los problemas de la vida, desde los más nimios a los devastadores.

Un niño pequeño tiene excusa para no pensar en la amenaza nuclear, la degradación del medio ambiente, secuestros, injusticias, violaciones, etc. Pero como padres hay que pensar en todo eso, a menos que estemos seguros de que su ignorancia temporal no termina en tragedia.

En el negocio del software, se suele equiparar madurez con excelencia técnica (CMM). Lo que se necesita ahora, sin embargo, es considerar la madurez en su sentido tradicional.

La Gestión de Riesgos es el proceso de pensar en acciones correctivas antes de que los problemas ocurran, mientras son meras abstracciones

Un riesgo es un evento futuro posible que produciría un resultado no deseado. También suele emplearse la palabra riesgo para designar al efecto mismo no deseado, en lugar de la causa.

A veces se usa esta definición circular, muy gráfica: Un riesgo es un problema que aún no ha ocurrido. Un problema es un riesgo que se ha materializado.

Lo contrario a Gestión de Riesgos se llama Gestión de Crisis, y consiste en intentar descubrir qué hacer con los problemas después de que ocurren.

¿Qué no es Gestión de Riesgos?

A veces, las empresas que se dicen "proclives al riesgo" son víctimas de un extraño comportamiento: tratan de enfatizar el pensamiento positivo ignorando las posibles

consecuencias negativas de los riesgos que asumen. Pero nadie es tan tonto como para ignorar todos los riesgos, sino que lo hacen selectivamente. Tienen un cuidado exquisito para listar, analizar y monitorizar todos los riesgos menores (los que se pueden compensar mediante la gestión), y sólo ignoran los realmente desagradables.

La gente se guarda de no tropezar con los raíles, pero ¡no ven el tren que se acerca!

Algunas organizaciones padecen esta “miopía selectiva” en Gestión de Riesgos: sólo ven las pequeñas preocupaciones (de fácil solución), pero no las pesadillas (especificaciones no aceptadas, hitos críticos que no se van a cumplir, subcontratistas que desaparecen, etc.). Esto definitivamente no es Gestión de Riesgos.

Gestión de Riesgos tampoco es lo mismo que “preocuparse por el proyecto”.

El caso del aeropuerto de Denver

En 1988 se decidió reemplazar el aeropuerto de Denver: el nuevo aeropuerto reduciría costes, ruido y polución. Eliminaría retrasos y permitiría el crecimiento de la ciudad. La fecha estimada de apertura era el 31/10/93. Finalmente, se abrió parcialmente en 1995, después de unas pérdidas por retrasos de unos 500 millones de dólares. ¿Qué pasó?

En la fecha prevista, todo estaba listo menos un sistema software llamado Sistema Automático de Gestión de Equipajes (ABHS).

Un artículo en Scientific American achacaba el retraso al proceso software, y proponía como solución aumentar nivel CMM. Pero un proceso software mejorado no elimina las incertidumbres en los proyectos. Además, un análisis en profundidad habría descubierto razones de otro tipo:

Pregunta: ¿Por qué no pudo abrir el aeropuerto sin el ABHS?

Respuesta: El software estaba en el camino crítico. Sin este software, no se podía atender a los pasajeros ni un solo día.

P.: ¿Por qué ABHS estaba en el camino crítico?

R.: Porque no se podía mover el equipaje sin el sistema: Tele-carritos, lectores de código de barras, dispositivos de escaneo, descargadores automáticos...

P.: ¿No hay más alternativas para mover el equipaje?

R.: Sí. Mozos empujando carritos y el método tradicional en los aeropuertos de cintas transportadoras y camiones pequeños.

P.: Si ABHS no estaba listo a tiempo, ¿por qué no se usaron los métodos alternativos?

R.: Los túneles de los telecarritos eran muy bajos.

P.: ¿No se podían rediseñar los túneles?

R.: Sí, pero no había tiempo. Cuando se descubrió que ABHS se retrasaba, ya estaban construidos, y el tiempo de rehabilitación se estimó mayor que el necesario para perfeccionar el software.

P.: ¿No podían empezar antes las obras de rehabilitación?

R.: Sí, pero se juzgó inapropiado. El tiempo y el dinero se malgastarían si el software terminaba a tiempo, como aseguraba la dirección.

P.: ¿No se vio el retraso de ABHS como un riesgo potencial?

R.: Sólo después de que ocurrió. Antes de ello: Calendario agresivo y gestión optimista.

P.: ¿No es frecuente que se retrasen los proyectos software?

R.: Sí, pero se suponía que éste iba a ser diferente.

P.: ¿Se sabía de alguna experiencia previa similar?

R.: El aeropuerto de Munich había instalado un piloto de ABHS parecido.

P.: ¿El equipo visitó el proyecto de Munich? Si es así, ¿qué aprendió?

R.: Sí. El equipo de Munich asignó 2 años de pruebas y convivencia con el sistema antiguo durante 6 meses para afinar el sistema nuevo. Aconsejaron hacer lo mismo.

P.: ¿La dirección del proyecto siguió este consejo?

R.: No había tiempo.

P.: ¿El equipo avisó suficientemente del retraso inminente?

R.: Ya los licitantes advirtieron del riesgo en sus propuestas. Se contrató a BAE Automated Systems porque ofrecían menor plazo. Durante la ejecución, el proveedor advirtió repetidas veces del potencial retraso, al introducirse cada mes nuevos cambios. Todas las partes eran conscientes de que se trataba de hacer un proyecto de 4 años en 2. Todas estas evidencias fueron ignoradas.

Conclusión

El fracaso se debió a la Gestión de Riesgos, no al proceso software. Cualquier lista de riesgos habría incluido el retraso en la entrega del software como un riesgo significativo. El análisis de exposición al riesgo habría mostrado que como ABHS estaba en el camino crítico, cualquier retraso retrasaría la apertura del aeropuerto, resultando una penalización de 33 millones de dólares al mes. La estrategia de mitigación evidente habría sido mover ABHS fuera del camino crítico. Unos pocos millones de dólares gastados en un esquema alternativo de gestión de equipajes habrían ahorrado 500 millones de los contribuyentes.

¿Quién tuvo la culpa? No el proveedor de software. Además, seguro que no fue el único que se retrasó. En Gestión de Riesgos siempre tiene la culpa el responsable de pagar las consecuencias de los riesgos que ignora. En este caso, la Administración local de Denver.

¿Por qué no?

A continuación se reproducen las seis razones habitualmente esgrimidas por el equipo de dirección en contra de la gestión de riesgos:

1. **El cliente no tiene madurez suficiente para enfrentarse a los riesgos:** “Si les decimos la verdad, los responsables tendrían miedo de hacer el proyecto, así que hay que mentirles...”. Pero los clientes actuales de IT tienen más poder, saben más y tienen buena memoria. Ya están acostumbrados al riesgo fuera de IT (¡y no les gustan las mentiras!).
2. **Demasiada incertidumbre:** “Puedo decir que habrá un margen de error sobre la fecha estimada, ¡pero no tan grande! ...”. Pero en IT las estimaciones son del tipo “la entrega se producirá entre el mes 18 y el 29, con un 85% de confianza de que sea el 24”, hay que vivir con ello.
3. **Explicitar la ventana de incertidumbre excusa el mal rendimiento:** “Si les digo a nuestros programadores que terminen el trabajo entre julio y diciembre, se dormirán...”. Pero objetivos y estimaciones no tienen por qué coincidir: utiliza los objetivos para incentivar el rendimiento. A la hora de hacer promesas a los clientes y a los jefes utiliza las estimaciones.
4. **Es mejor utilizar un enfoque de “gestión para el éxito”:** “No hacemos Gestión de Riesgos, pero los vigilamos y gestionamos para que no ocurran...”. Pero los riesgos se gestionan, no se anulan así como así. Hay riesgos intrínsecos que sólo se evitan renunciando a parte del producto.
5. **No hay datos suficientes para realizar una gestión de riesgos efectiva.** “No sabemos

suficiente sobre los riesgos que pueden afectar a este proyecto...". Pero los riesgos más importantes en los proyectos de IT son siempre los mismos.

6. **La gestión de riesgos es peligrosa si se realiza aisladamente.** "Honestamente, no quiero ser el único que haga Gestión de Riesgos." Tiene razón: no tiene sentido que un jefe de proyectos haga Gestión de Riesgos si está rodeado de compañeros que hacen gestión optimista.

El impacto cultural

Introducir la práctica de la Gestión de Riesgos en una organización es un gran reto de dirección. En casi todas las organizaciones existen barreras culturales que desincentivan cualquier empeño por gestionar riesgos en los proyectos de software. He aquí algunas de esas barreras:

- La actitud de "se puede hacer".
- El deseo de no defraudar.
- El imperativo de preservar el color rosa del escenario más color de rosa.
- Miedo a expresar la incertidumbre.
- La necesidad de aparentar que la situación está bajo control (incluso cuando es ilusorio).
- La tentación para usar el poder político para disfrazar la realidad.
- El pensamiento a corto plazo, tan común en las actividades humanas.
- Se penalizan las predicciones poco atractivas, pero no los resultados poco atractivos.
- Prometer a lo grande es más importante que entregar.
- Está permitido equivocarse, pero no se permite expresar incertidumbres.

Especialmente difícil es la fase inicial de la Gestión de Riesgos (que luego ha de mantenerse continuamente) que trata de descubrir los riesgos. Aquí son preocupantes las reglas no escritas que desincentivan la comunicación:

- No seas negativo.
- No eleves un problema, a menos que tengas la solución.
- No digas que algo es un problema si no puedes probar que lo es.
- No seas aguafiestas.
- No describas un problema a menos que quieras convertirte en el responsable de su solución inmediata.

¿Cómo se hace?

Las actividades básicas que deben seguirse en Gestión de Riesgos son:

1. Descubrimiento de riesgos

Esta fase supone vencer las barreras culturales que dificultan la comunicación de riesgos, ya comentadas. El equipo encargado (no debe hacerse individualmente) debe concederse la licencia temporal de "pensar negativamente". Esto exige "cambiar el chip", y por fortuna existen algunas técnicas que sirven de ayuda:

- **Brainstorming sobre catástrofes:** hay que imaginar resultados catastróficos del proyecto. El facilitador puede utilizar ciertas preguntas-truco como: ¿cuáles son tus peores miedos, o tu peor pesadilla? ¿Qué nos puede llevar a salir en los periódicos por lo mal que lo hemos hecho? Describe el mejor resultado del proyecto y después dale la vuelta. ¿Hay desastres que no sean culpa de nadie? ¿Que sean culpa nuestra? ¿Del cliente? ¿De dirección? ¿Tuya? ¿Puede darse un éxito generalizado, pero que no contente a algún

interesado?

- Construcción de escenarios: describir los escenarios que conducen a las situaciones catastróficas comentadas. Asignar probabilidades de ocurrencia.
- Análisis de las causas: Normalmente, un riesgo podrá identificarse con la causa que produce un escenario que deriva en un resultado catastrófico. Esta fase es la más compleja, y puede requerir la involucración de expertos.

Así pueden descubrirse los riesgos específicos del proyecto en cuestión. A éstos hay que añadir los que se deducen de la historia pasada en proyectos parecidos (un problema de ayer es un riesgo de hoy) y los riesgos habituales en proyectos software, de dominio público.

2. Análisis del grado de exposición

La exposición de un riesgo es resultado de multiplicar su probabilidad por el impacto económico.

3. Plan de respuesta al riesgo

Una vez que tenemos la lista de riesgos, ¿qué hacemos con ella? Todo riesgo es una molestia, y habrá presión para quitarlo. Afortunadamente, los riesgos caducan (e.g. el riesgo de entrega expira la fecha de entrega, el riesgo de no-conformidad expira con la aceptación, el riesgo de “especificación imposible” expira cuando se cierra la especificación, al cierre del proyecto no hay ningún riesgo...).

Pero ¿qué hacer antes de que expire un riesgo? Se pueden planear 4 respuestas diferentes:

- Evitar: no hacer el proyecto o la parte que supone ese riesgo.
- Contener: reservar tiempo y dinero por si ocurre.
- Mitigar: prevención y acción correctora para reducir el efecto.
- Evadir: cruzar los dedos para que no pase. Aquí la pregunta es: ¿Tienes derecho a creer que no pasará?

Hay un tipo de riesgos tan graves que si ocurren paralizan el proyecto (e.g. la competencia se adelanta). La gestión de los mismos corresponde a un nivel superior al director del proyecto. Se han de tomar como asunciones.

4. Mitigación

Cuando se identifica un riesgo, el plan de respuesta más común suele ser la mitigación. La mitigación tiene dos partes:

- Los pasos que hay que dar antes de que el riesgo se materialice para minimizar el impacto de las acciones correctoras.
- Las acciones correctoras que se ejecutan después de su materialización.

Para detectar cuanto antes la materialización de un riesgo, o lo que es lo mismo, la transición de un riesgo a un problema, suelen considerarse indicadores, señales, o disparadores, que hay que estar continuamente monitorizando.

Por ejemplo, en un juicio con jurado, un riesgo puede ser que un miembro del jurado caiga enfermo, con lo que el juicio debería aplazarse. El plan de mitigación es designar suplentes y activar rápidamente la sustitución cuando el miembro del jurado declara su incapacidad por enfermedad.

Otro riesgo, más próximo a los proyectos software, es la rotación del personal. El plan de mitigación podría ser sobredimensionar el equipo. El disparador sería la renuncia del programador (aunque aquí hay más síntomas que un buen jefe de equipo debería

monitorizar). La acción correctora sería la sustitución por otro programador, con tiempo razonable para la transmisión del conocimiento.

Como puede verse, en todos los casos, la mitigación cuesta tiempo y dinero.

5. Monitorización continua

La supervisión y control de riesgos ha de realizarse de manera continua durante la ejecución del proyecto. Fundamentalmente hay que monitorizar los indicadores de transición, o disparadores, pero además es necesario que el proceso de descubrimiento continúe, por si aparecen nuevos riesgos, y también hay que monitorizar cuando expiran.

En este sentido, algo especialmente importante en los proyectos de software es monitorizar el cierre de las especificaciones y el avance de los resultados. Lo ideal es aplicar métricas en ambos casos:

- Las métricas de cierre de especificaciones tienen que ver con los requisitos y los cambios. Cuando estas métricas demuestran que la especificación se puede dar por finalizada, expira el riesgo de “especificación imposible”, principal causa de cancelación de proyectos software.
- Las métricas de seguimiento del esfuerzo en proyectos software más conocidas se obtienen a partir de la aplicación del método de Gestión del Valor Ganado (EVM), que en proyectos software significa practicar iteraciones incrementales o entregas parciales, e ir midiendo el porcentaje completado (utilizando métricas de puntos función o de esfuerzo relativo).

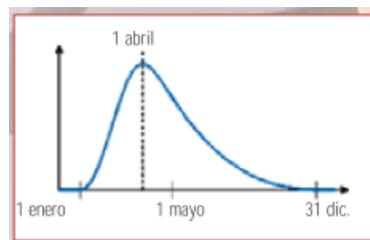
Diagramas de Riesgo

Eres el jefe de equipo a cargo de un proyecto. La fecha de entrega planificada es el 30 de octubre. Es junio y ya sabes que no entregarás en fecha, pero nada más. Viene un experto en métricas de software para ayudarte con tus previsiones. Diría estas cosas, por ejemplo:

- Terminar antes de enero es imposible.
- La fecha más probable para entregar un producto aceptable es el 1 de abril, pero incluso esta fecha no es muy creíble (30%).
- Te vas a mayo si quieres publicar un deadline con el 50% de confianza.
- Si quieres probabilidad de retraso virtualmente nula, debes publicar como fecha de fin de proyecto el 31 de diciembre.

Toda esta información se puede expresar en un diagrama de riesgo, que sirve para cuantificar la ventana de incertidumbre:

Estos diagramas indican la probabilidad relativa. La probabilidad absoluta (de entregar antes de una fecha) es el porcentaje de área bajo la curva a la izquierda. Como puede verse, la fecha más temprana (que suele plantearse como objetivo) es imposible (el área que encierra es cero). Se puede combinar el efecto de varios riesgos en un diagrama utilizando técnicas de simulación de Monte Carlo.



Riesgos habituales en proyectos software

Éstas son las 10 principales fuentes de riesgo en proyectos software:

1. Requisitos: ¿qué es exactamente lo que el sistema tiene que hacer?
2. Adecuación: ¿cómo interactuará el sistema con los operadores humanos y otros sistemas?
3. Entorno cambiante: ¿cómo cambiarán las necesidades y los objetivos durante el período de

ejecución?

4. Recursos: ¿qué habilidades humanas clave estarán disponibles (cuando se necesiten) según avance el proyecto?

5. Dirección: ¿tendrá el equipo directivo el talento suficiente para favorecer la creación de equipos productivos, mantener la moral elevada, la rotación baja y coordinar conjuntos complejos de tareas interrelacionadas?

6. Cadena de suministros: el desempeño de terceros, ¿será el esperado?

7. Política: ¿cuál será el efecto del uso del poder político para disfrazar la realidad e imponer restricciones inconsistentes con la finalización exitosa del proyecto?

8. Conflictos: ¿cómo resolverán los miembros de una comunidad diversa de interesados sus objetivos mutuamente incompatibles?

9. Innovación: ¿cómo afectarán al resultado final las tecnologías y enfoques únicos a este proyecto?

10. Escala: ¿cómo impactará en el rendimiento del proyecto la superación de la experiencia pasada en volumen y alcance?

Si bien estas fuentes de riesgo son muy frecuentes, no se han publicado suficientes datos como para sistematizar su impacto. Todo lo contrario ha ocurrido con los siguientes 5 riesgos aplicables a proyectos software⁶. A efectos prácticos, estos riesgos se suelen considerar como factores de corrección en las estimaciones:

1. Errores de estimación: Se suelen estimar plazos de ejecución poco realistas, típicamente un 18% por debajo. Los mejores esfuerzos de sizing rebajan el factor sólo hasta un 15%.
2. Inflación de requisitos: La naturaleza cambiante de los proyectos software hace que los requisitos cambien durante la ejecución, en mayor o menor medida. Todo cambio, aunque sea para eliminar una parte ya construida, significa mayor esfuerzo (típicamente un 7% sobre lo estimado).
3. Rotación del personal: La rotación del equipo de desarrollo afecta también a las estimaciones (en media, un factor del 4%).
4. Productividad: un equipo poco cohesionado puede retrasar la finalización típicamente hasta un 15% (pero el efecto contrario puede producirse con equipos muy cohesionados: el todo a veces es mejor que la suma de las partes).
5. Especificación imposible: uno de cada 7 proyectos se cancelan anticipadamente. La causa más común suele ser la imposibilidad de cerrar los requisitos. Se podría pensar que este riesgo no es preocupante, porque si no hay acuerdo en los requisitos, el proyecto se cancela y el daño para las partes es limitado. Por desgracia, lo que suele ocurrir es que los interesados fingen cooperar pero no se ponen de acuerdo sobre la especificación. Es posible especificar ambiguamente, pero no es posible codificar ambiguamente. Cuando el problema aparece en la fase de construcción, ya es demasiado tarde, y el soporte de los interesados está muy erosionado. Otras veces, suele ocurrir un fenómeno curioso, que se ha dado en llamar “contra-implementación”: un grupo contrario al proyecto decide sobrecargar la especificación para que el beneficio del proyecto sea muy inferior a su coste (las funcionalidades de la A a la F son necesarias y justifican el proyecto, pero este grupo de interesados aparentemente entusiasta añade de la G a la Z).

⁶ Estudio sobre proyectos de tamaño inferior a 3.000 puntos función, con plazo de ejecución de 1-2 años y con equipos inferiores a 10 personas.

¿Quién lo hace?

He aquí un checklist con las actividades que debería practicar una organización que quiera decir, con total propiedad, que hace Gestión de Riesgos:

1. Para cada proyecto existe un censo de riesgos, incluyendo los riesgos habituales en proyectos software. Los riesgos tienen naturaleza de causa, más que de efecto.
2. Hay en marcha un proceso continuo de descubrimiento de riesgos. Se trata de una actividad transparente y no se rechazan las contribuciones de nuevos participantes. Se dan pasos específicos para asegurar la captura de ideas desagradables. Puede que incluso haya un canal anónimo para facilitar que la gente comunique malas noticias.
3. Se usan diagramas de incertidumbre para cuantificar riesgos simples y agregados. La cultura organizativa considera poco profesional hacer compromisos sin considerar explícitamente la incertidumbre.
4. En los proyectos, las fechas objetivo no coinciden con las estimaciones, que son más conservadoras (e.g. si la fecha objetivo son 12 meses para una entrega, la estimación dirá 18). El nivel de confianza de estimaciones y objetivos se expresa con diagramas de incertidumbre.
5. Cada riesgo tiene asociado un disparador. Se practica una monitorización continua de los disparadores, para detectar cuanto antes la materialización de los riesgos.
6. Cada riesgo tiene asociado un plan de contingencia y un plan de mitigación. Las actividades de contingencia forman parte del alcance, como tareas posibles. Las actividades de mitigación se incluyen como definitivas y se ejecutan mucho antes que las señales de alarma.
7. Para cada riesgo se cuantifica su exposición, en términos económicos.
8. Se cuantifica el beneficio futuro del proyecto (el valor que tendrá una vez terminado). Se priorizan los componentes del sistema conforme a su contribución al valor final, y esta ordenación es tenida en cuenta en el plan de entregas.
9. Se practica algún grado de incrementalismo en el plan del proyecto. Algunas versiones son entregadas efectivamente a los interesados o bien son pseudo-entregadas (se representan todos los pasos necesarios pero sin hacer la entrega real). Sobre cada entrega completada, se miden tiempos, esfuerzos y tamaño relativo, para poder cuantificar el nivel de completitud hasta la fecha.

Este artículo está basado en el libro: *Waltzing with Bears: Managing Risk on Software Projects*. Tom DeMarco & Timothy Lister. Dorset House Publishing, 2003



personas



procesos



tecnologías

Gestión de la incertidumbre en el proceso de estimación durante el ciclo de vida



Gestión de la incertidumbre en el proceso de estimación durante el ciclo de vida

Las medidas tienen significados y utilidades distintas en función del momento en que sean recogidas o analizadas. En esta ponencia se presentan los miedos y prejuicios de las organizaciones, que provocan el fracaso de las iniciativas de medida de tamaño. Se proponen soluciones de confianza basadas en una estimación progresiva durante todo el ciclo de vida, en la que la incertidumbre inicial se va reduciendo conforme se tiene más información del proyecto. Así, se pretende mostrar una visión práctica mediante la utilización de los repositorios de proyectos más representativos de la industria del software (ISBSG y QSM) y su aplicación en distintos momentos, desde la estimación temprana hasta la medida del progreso del proyecto. El primero, ISBSG, tiene carácter público y dispone de 2.000 proyectos suministrados voluntariamente por organizaciones de todo el mundo. El segundo ha sido elaborado por la compañía QSM a lo largo de los últimos 25 años, alcanzando cerca de 7.000 proyectos y es la base de la suite de herramientas SLIM de estimación, control y gestión de métricas.

1. El tamaño del software

1.1. La estimación de proyectos: base de una planificación realista

La crisis del software se ha convertido en un tópico durante los últimos 30 o 40 años. Las líneas de tendencia de costes del hardware y del software divergen con el paso del tiempo, lo que agrava aún más la situación.

En la actualidad, los responsables de tecnologías de la información se encuentran con que la experiencia de los usuarios ha aumentado: están acostumbrados al uso del ordenador personal y de paquetes integrados que les ofrecen soluciones rápidas a sus necesidades. Estos usuarios expertos demandan la construcción de aplicaciones cada vez más complejas y en menores plazos de tiempo. Como consecuencia, aumentan los costes de desarrollo del software a la vez que la insatisfacción de los usuarios, que ven que sus necesidades se acaban resolviendo tarde y con poca calidad.

El análisis de estos retrasos nos lleva a la conclusión de que las desviaciones sobre la planificación suelen tener su origen en una estimación deficiente y poca realista del esfuerzo y duración necesarios para realizar un proyecto.

Desde finales de los años 80 se han publicado diversos modelos de estimación de proyectos, con el objeto de servir de base para realizar una planificación realista, normalmente basada en la experiencia de la organización. La solución para encontrar estimaciones

Autor: Ramiro Carballo Gutiérrez director Comercial de GESEIN.

Publicado en la Revista CALIDAD, marzo abril de 2007

Las medidas tienen significados y utilidades distintas en función del momento en que sean recogidas o analizadas. En esta ponencia se presentan los miedos y prejuicios de las organizaciones, que provocan el fracaso de las iniciativas de medida de tamaño. Se proponen soluciones de confianza basadas en una estimación progresiva durante todo el ciclo de vida, en la que la incertidumbre inicial se va reduciendo conforme se tiene más información del proyecto. Así, se pretende mostrar una visión práctica mediante la utilización de los repositorios de proyectos más representativos de la industria

del software (ISBSG y QSM) y su aplicación en distintos momentos, desde la estimación temprana hasta la medida del progreso del proyecto. El primero, ISBSG, tiene carácter público y dispone de 2.000 proyectos suministrados voluntariamente por organizaciones de todo el mundo. El segundo ha sido elaborado por la compañía QSM a lo largo de los últimos 25 años, alcanzando cerca de 7.000 proyectos y es la base de la suite de herramientas SLIM de estimación, control y gestión de métricas.

1. El tamaño del software

1.1. La estimación de proyectos: base de una planificación realista

La crisis del software se ha convertido en un tópico durante los últimos 30 o 40 años. Las líneas de tendencia de costes del hardware y del software divergen con el paso del tiempo, lo que agrava aún más la situación.

En la actualidad, los responsables de tecnologías de la información se encuentran con que la experiencia de los usuarios ha aumentado: están acostumbrados al uso del ordenador personal y de paquetes integrados que les ofrecen soluciones rápidas a sus necesidades. Estos usuarios expertos demandan la construcción de aplicaciones cada vez más complejas y en menores plazos de tiempo. Como consecuencia, aumentan los costes de desarrollo del software a la vez que la insatisfacción de los usuarios, que ven que sus necesidades se acaban resolviendo tarde y con poca calidad.

El análisis de estos retrasos nos lleva a la conclusión de que las desviaciones sobre la

planificación suelen tener su origen en una estimación deficiente y poco realista del esfuerzo y duración necesarios para realizar un proyecto.

Desde finales de los años 70 se han publicado diversos modelos de estimación de proyectos, con el objeto de servir de base para realizar una planificación realista, normalmente basada en la experiencia de la organización. La solución para encontrar estimaciones realistas pasa por apoyarse en la experiencia de proyectos similares, recopilados en un repositorio.

1.2. El tamaño: factor clave de la estimación de proyectos

Si analizamos algunos de estos modelos para la estimación de proyectos, llegamos a la conclusión de que el tamaño es uno de los factores más determinantes en una estimación de tiempo y esfuerzo.

Por ejemplo, vemos cómo en la fórmula para la estimación del esfuerzo de COCOMO 2.0 [8] [9] el tamaño aparece elevado a los factores de escala (B), multiplicado por una constante (A) y por los multiplicadores de esfuerzo (EM):

$$PM_{\text{estimado}} = (\mu EM_i) * A * (\text{Tamaño})^B$$

Si analizamos la ecuación para la estimación de esfuerzo mediante comparación, propuesta por ISBSG para utilizar con su repositorio, el esfuerzo es resultado del producto de la productividad PDR (Project Delivery Rate) por el tamaño, más un componente de desplazamiento en función del entorno del proyecto.

$$\text{Esfuerzo} = \text{PDR} * \text{Tamaño} + \text{OHD}$$

No cabe duda de que, para estimar un proyecto, se debe estimar previamente el tamaño. Parece razonable por tratarse de la única referencia que tenemos de la cantidad de trabajo que se debe realizar. Para estimar el tamaño, se debe partir de la información disponible acerca del proyecto en sí. La primera impresión es que esta tarea no es precisamente sencilla.

1.3. Percepción: el tamaño no se puede estimar

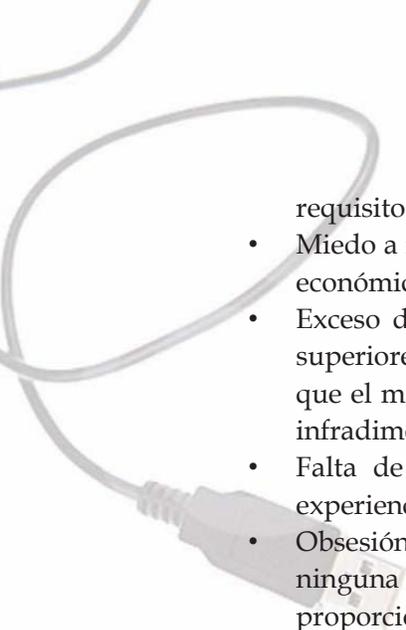
Larry Putnam de QSM, autor del modelo SLIM (Software Lifecycle Model), cuenta en su libro [1] una anécdota en relación con la medida del tamaño del software: un programador comenta

“No se puede medir la materia del pensamiento puro”, a lo que el responsable de la Oficina de Proyectos contesta: “Puede que no, aunque nosotros solíamos pesar la caja de las tarjetas perforadas que contienen el programa”. Escéptico, el programador se queja: “¡El programa son los agujeros... y no pesan nada!”.

La verdad es que si se pueden pesar o contar las tarjetas, equivalente a contar las líneas de código en la actualidad, es porque el programa ya está acabado. Si estamos hablando de estimación de proyectos, no se dispondrá de esta facilidad, y nuestra meta será estimar el tamaño de la funcionalidad que hay que desarrollar para conocer el esfuerzo que requerirá.

La sensación que se tiene cuando se intenta estimar lo que costará realizar una tarea intelectual, como el desarrollo de software, es que no se puede saber con exactitud hasta que no se está realizando. De hecho, la historia del desarrollo de software nos enseña que los intentos de medir el tamaño de un sistema fracasan en un alto porcentaje. Si analizamos los motivos, nos encontramos que los fracasos se atribuyen habitualmente a motivos humanos y no a la utilización de unas u otras unidades de medida, como se suele achacar. Por ejemplo:

- Falta de definición del proyecto, no conocer cuáles son los requisitos del sistema en las fases tempranas del proyecto. En fases posteriores, el problema es la volatilidad de estos



requisitos.

- Miedo a realizar predicciones en las que existe un alto riesgo de equivocarse sobre temas económicamente muy importantes.
- Exceso de optimismo en los gestores, que enmascara la intención de complacer a los superiores o de evitar enfrentamientos. El propio Boehm, autor de COCOMO [3], dijo que el mayor impedimento para hacer estimaciones de coste precisas es el problema del infradimensionamiento del tamaño del software.
- Falta de registros apropiados para asociar la estimación de nuevos proyectos a la experiencia de proyectos anteriores.
- Obsesión por presentar una cifra exacta como resultado de la estimación. No existe ninguna bola de cristal que sirva para adivinar en lugar de estimar. Lo adecuado es proporcionar un rango de valores con una distribución de probabilidad.

Esta situación debe ser reconsiderada por las organizaciones de desarrollo de software: la estimación de tamaño es un problema difícil, pero no imposible. Lo que hay que conseguir es estimar mejor y, para ello, hay que hacer que las personas se encuentren bien con la manera de hacer las estimaciones:

- Hay que asumir el reto de estimar con requisitos incompletos y reestimar conforme se van completando o cambiando.
- Es necesario poner los medios para tener datos históricos, bien organizados y accesibles para gestores y para estimadores.
- El estimador debe aislarse de las tendencias pesimistas u optimistas. Los partidarios de estas tendencias se verán reconocidos en la interpretación del rango de incertidumbre con el que se proporcione el resultado de la estimación.
- La incertidumbre debe acompañar todo el proceso de estimación y se debe ir reduciendo conforme se avanza en el ciclo de vida con más detalles sobre el proyecto. Esta incertidumbre se transmitirá a los otros valores que sean estimados en función del tamaño (tiempo, esfuerzo, etc.). La incertidumbre deja de existir sólo en los proyectos finalizados, donde podemos dar un número como resultado.
- A la vista del punto anterior, el proceso de estimación se debe considerar un proceso continuo.

2. Cuándo medir el tamaño

2.1. A lo largo de todo el ciclo de vida

Cuanto antes se cuantifique un proyecto, antes lo tendremos bajo control. Sin embargo, en las fases tempranas no tenemos demasiada información. Tradicionalmente se ha dicho que la información necesaria para medir el tamaño en puntos función está disponible por primera vez en la fase de diseño. No es demasiado tarde para ser útil, ya que los costes más importantes ocurren durante las fases de construcción y pruebas, pero sería más útil si nuestro proceso de estimación se iniciase antes.

Conocida la necesidad de estimar el tamaño desde las fases tempranas del ciclo de vida, el Manual de Medida de Puntos Función de IFPUG [5] proporciona reglas para medir una vez completados los requisitos. Desde el punto de vista práctico, existen heurísticas que permiten estimar aun antes.

De esta manera, se pretende disponer cuanto antes de una estimación inicial, válida dentro de un rango e ir disminuyendo la incertidumbre conforme el proyecto progresa y se consigue

más información que permite hacer estimaciones más acertadas.

En resumen, se debe medir el tamaño de un proyecto pronto y a menudo. IFPUG recomienda en su “Guía para la medida del software” [4] medir al menos en las siguientes ocasiones:

- Al inicio del proyecto.
- Al final de la fase de análisis de requisitos.
- Al final de la fase de diseño.
- Cada vez que se produzca un cambio en el alcance del proyecto.

Si se tiene en cuenta que entre estos hitos también se pueden realizar estimaciones de manera aproximada, los puntos siguientes describen los resultados que se pueden obtener en función de la información disponible en cada fase del proyecto.

2.2. Estimación de tamaño durante el estudio de viabilidad del proyecto

El estudio de viabilidad debería ser la primera fase del ciclo de vida. Como un objetivo de esta fase, se puede decidir no iniciar el proyecto, y para ello es imprescindible contar con algún tipo de estimación. Sin embargo, no existe suficiente información acerca de la funcionalidad que se desea desarrollar para solucionar el problema que hace que el proyecto sea necesario. Por lo tanto, no se puede realizar medida de puntos función.

Habitualmente en esta fase se busca en la historia de la organización la existencia de proyectos similares en cuanto al tipo de problema que pretendían solucionar y su alcance, asumiendo que ambos proyectos deben tener un tamaño similar.

Larry Putnam propone hacer una estimación basada en lógica difusa, de manera que, según la experiencia del estimador y la existencia de proyectos similares en el repositorio de QSM, se considera que el tamaño será “muy pequeño”, “pequeño”, “medio” o “muy grande”. Para cada uno de estos tamaños difusos, existe un rango de valores de tamaños habituales que se puede utilizar para una estimación inicial.

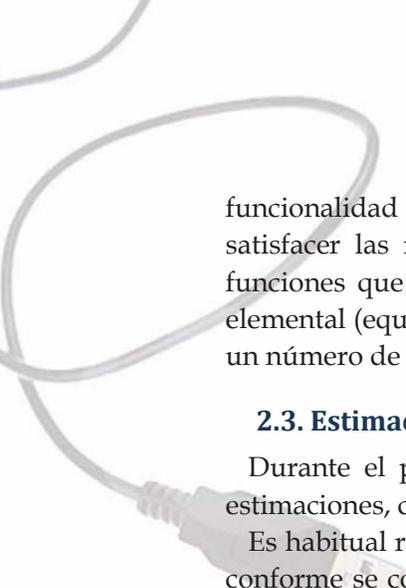
Otras técnicas consisten en llevar a cabo el ejercicio de suponer un modelo conceptual de datos, de alto nivel, implícito en lo que conocemos del proyecto y completarlo con aquella parte del modelo de datos que se espera necesaria si se compara con otros proyectos similares. Sobre este modelo conceptual de datos de alto nivel, se puede aplicar alguna de las heurísticas

para estimar el tamaño, basadas en el número de entidades del modelo, de donde se puede deducir el número estimado de puntos función del proyecto.

Un método bastante difundido es el desarrollado

por la consultora italiana DPO, denominado “Early Function Points” (EFP). El método estima el número de puntos función del proyecto mediante la clasificación de la información disponible entre distintos tipos de elementos que describen funcionalidad o ficheros de datos. Por ejemplo, un elemento de funcionalidad puede ser de cuatro tipos, de mayor a menor tamaño: macrofuncionalidad (puede ser un pequeño desarrollo o un subsistema),

LD (datos lógicos)	MÍN.	MED.	MÁX.	Microfunción	MÍN.	MED.	MÁX.
Simple	5	6	7	Complejidad única	16	18	20
Medio	8	9	10	Función	MÍN.	MED.	MÁX.
Complejo	13	14	15	Pequeña	45	56	67
Baja multiplicidad	14	18	22	Media	73	91	109
Alta multiplicidad	27	39	51	Grande	106	133	160
Función primitiva	MÍN.	MED.	MÁX.	Macrofunción	MÍN.	MED.	MÁX.
Pequeña	4	5	7	Pequeña	151	215	280
Media	5	6	8	Media	302	431	560
Grande	4	5	7	Grande	603	861	1119



funcionalidad (como un conjunto de funciones operativas, que son imprescindibles para satisfacer las necesidades del usuario), microfuncionalidad (equivalente a un conjunto de funciones que hacen gestión completa de un fichero de datos) y funcionalidad primitiva o elemental (equivalente a las definidas por IFPUG). Cada uno de estos elementos tiene asociado un número de puntos función, determinado estadísticamente según la experiencia.

2.3. Estimación de tamaño durante la recogida de los requisitos

Durante el proceso en el que se van definiendo los requisitos es posible realizar varias estimaciones, consiguiendo más precisión en cada una de ellas.

Es habitual realizar un modelo conceptual de datos para ayudar a comprender los requisitos conforme se completan éstos. Este modelo no necesita tener atributos definidos ni cumplir la tercera forma normal, pero es suficiente para analizar qué entidades requerirán procesos para ser gestionadas (altas, baja, modificación, consultas). Si se ha definido un diagrama de contexto, se dispondrá de la mayor parte de interacciones del sistema con el usuario o con otros sistemas.

En este momento se puede hacer una estimación de puntos función asumiendo que todas las transacciones y ficheros tienen una complejidad media.

2.4. Estimación de tamaño con los requisitos completamente definidos

En este momento se puede hacer una medida exacta de los puntos función del proyecto.

El modelo de datos se ha completado y tiene todos sus atributos definidos, así como la funcionalidad en la que se ha determinado cada interacción, informe o pantalla, y la información asociada.

A partir de este momento se realizarán nuevas estimaciones en las sucesivas fases del ciclo de vida del proyecto, que proporcionarán una visión de seguimiento del mismo fundamentalmente en los aspectos de estabilidad de requisitos, negociación sobre los cambios, valor aportado al negocio, etc.

2.5. El tamaño en la fase de diseño

Con más información que en la fase de análisis de requisitos, durante la fase de diseño se pueden medir los puntos función del proyecto. En teoría no debemos encontrar diferencias con la última medida realizada una vez completados los requisitos, excepto que éstos se hayan visto alterados.

Éste es el momento de empezar a tener en cuenta el impacto de los cambios en el proyecto. Se puede tratar de un incremento de la funcionalidad, aumentando el alcance que se definió en fases anteriores. En tal caso, la medida actual de puntos función debe ser superior a las realizadas anteriormente.

En el caso de que los puntos función sean de mejora (“enhancement function points”), no existe alteración en el número de puntos función medidos en esta fase respecto a las anteriores, debido a que las funcionalidades han sido sustituidas por otras, sin aumentar ni disminuir. Existe un alto riesgo de que el impacto de estos cambios sea ignorado.

2.6. El tamaño al final de las pruebas de aceptación

No deberían existir cambios entre la fase de diseño y las pruebas de aceptación del cliente, es decir, durante las fases de implementación y pruebas. Sin embargo, éste suele ser el periodo en el que se negocia más habitualmente la funcionalidad frente al calendario y los costes, entre el

usuario y el jefe del proyecto. Los puntos función pueden ser un valor objetivo que sirva como referencia para cuantificar estas negociaciones.

2.7. El tamaño al final del proyecto

Tiene sentido actualizar la medida de puntos función al final del proyecto. Un estudio post mortem ayudará a realizar estimaciones de nuevos proyectos con más precisión en el futuro. Además, determina el nivel de soporte que requerirá el proyecto y permitirá decidir si merece la pena, teniendo en cuenta el valor que aporta al negocio como sistema de información.

3. La precisión de la estimación del tamaño

3.1. Calidad de la documentación y tiempo disponible

Según la consultora australiana Total Metrics [7], el nivel de detalle necesario para una medida de tamaño depende del propósito para el que se esté midiendo. Cada propósito requerirá diferentes niveles de calidad en la documentación disponible sobre el proyecto, pero también requerirá un tiempo para realizar la medida del que posiblemente no se disponga.

Total Metrics define seis niveles, desde el más al menos detallado:

- Nivel 1: Medida detallada, enlazada y etiquetada. No existe ninguna ambigüedad en la documentación. Se proporciona una precisión en la medida de +/- 10%. Se pueden medir 200 puntos función por jornada.
- Nivel 2: Medida detallada y enlazada. Precisión del +/- 10%. 250 puntos función por jornada.
- Nivel 3: Medida detallada. Precisión del +/- 10%. 300 puntos función por jornada.
- Nivel 4: Medida con complejidad por defecto. Precisión del +/- 15%. 400 puntos función por jornada.
- Nivel 5: Medida aproximada. Precisión del +/- 20 al 25%. 750 puntos función por jornada.
- Nivel 6: Estimación de tamaño aproximado, en base a correlaciones con la medida del número de entidades, casos de uso o informes. Precisión del +/- 20%. Se puede estimar el tamaño de una aplicación en una jornada.

En orden inverso, estos niveles tienen una correspondencia clara con las estimaciones de tamaño que se pueden realizar a lo largo del ciclo de vida. Como se ha comentado en el punto anterior, conforme progresa el proyecto, se dispone de más información, de manera que se consigue reducir la incertidumbre.

3.2. Estimación de tamaño basada en tres puntos

Antes se ha comentado la importancia de considerar una estimación de tamaño como un rango de posibles valores, en lugar de un único valor. La utilización de un único valor como resultado de la estimación se ha enumerado como uno de los motivos de fracaso de los intentos de estimación de tamaño en las organizaciones.

Dada la importancia que tiene la estimación de tamaño dentro del proceso de estimación de proyectos software, se ha establecido la necesidad de empezar a estimar desde las fases iniciales del proyecto, donde la información disponible es reducida.

Para solucionar los problemas de estimar pronto y gestionar la incertidumbre, se considera una buena práctica realizar una estimación de tamaño basada en tres puntos:

- El tamaño mínimo que se espera como posible.

- El tamaño más probable.
- El tamaño máximo que se espera en el peor caso.

Estos tres puntos determinan un rango de incertidumbre inicial, amplio en un principio, que se estrecha conforme el proyecto progresa y se consigue definir mejor.

Si se utilizase un repositorio de proyectos con medidas de tamaño que pudiesen utilizarse como referencia de tamaño en proyectos similares, se podría trabajar con la incertidumbre cuantificada en un valor de desviación típica en torno a un punto. Esto también se puede considerar en el caso en que se decida que distintas personas realicen diversas estimaciones del mismo proyecto, y se considere como resultado válido el rango determinado por la desviación típica en torno a un valor. Lo mismo se puede decir de un proyecto estimado por una persona con distintas técnicas de estimación, donde se ofrece como resultado la solución estadísticamente más probable y su desviación típica asociada.

La gestión de la incertidumbre en el tamaño tiene su impacto en la estimación del proyecto, trasladada como se ve a continuación.

4. La precisión de la estimación del proyecto

4.1. La incertidumbre según el modelo propuesto por ISBSG

“The International Software Benchmarking Standard Group” (ISBSG) es una asociación australiana que ha construido un repositorio de proyectos recopilados por todo el mundo.

En la actualidad, el repositorio contiene más de 2.000 proyectos, que son utilizados como base para la estimación. ISBSG propone unos métodos para realizar estimaciones aprovechando la información del repositorio.

En el caso de la técnica de macroestimación basada en comparaciones, se propone la selección de un grupo de proyectos del repositorio, que compartan características con el nuevo proyecto que se desea planificar. De este grupo se extrae el PDR, como la productividad estimada, enunciada en horas por punto función.

La ecuación para estimar el esfuerzo, descrita al inicio de este artículo, es la siguiente:

$$\text{Esfuerzo} = \text{PDR} * \text{Tamaño}$$

Dado que partimos de una productividad estimada estadísticamente desde el repositorio, es recomendable utilizar un rango de valores para PDR: por ejemplo, el valor esperado es 7,6, siendo el límite optimista 6,1 y el pesimista 8,9 horas por punto función.

Tabla 2: Ejemplo de estimación de esfuerzo con incertumbre

PDR	FP		
	440	550	687,5
6,1	2.684	3.355	4.194
7,6	3.344	4.180	5.225
8,9	3.916	4.895	6.119

Suponiendo que estamos realizando la estimación en una fase temprana del ciclo de vida, trabajamos con un rango de valores en torno a 550 puntos función. Del resultado de aplicar la ecuación con las distintas combinaciones de valores de tamaño y PDR, se obtiene la siguiente matriz:

A la vista de los resultados, sería aceptable proponer como solución de la estimación de esfuerzo el valor de 4.180 horas/hombre, matizando como intervalo probable el rango de 3.350 a 5.060 horas/hombre (+/- 20%). Si se quiere mantener un rango de probabilidad más seguro, se deben tener en cuenta los extremos, entre 2.684 y 6.119 horas/hombre (+/- 40% aprox.).

4.2. La incertidumbre según SLIM (modelo de Putnam) El modelo de Putnam genera estimaciones en función de los parámetros del tamaño, la productividad (PI) y el índice que relaciona la acumulación de personal en un proyecto, y que caracteriza la presión de tiempo (MBI, Manpower Build Up).

Tanto PI como MBI son deducidos estadísticamente del repositorio de QSM de 7.000 proyectos, buscando entre los de características similares al que se desea estimar, fundamentalmente en cuanto al orden de tamaño (las organizaciones son más productivas en proyectos de mayor tamaño) y tipo de aplicación (gestión, comunicaciones, tiempo real, etc.). Por este motivo estos dos indicadores también son gestionados con incertidumbre por el sistema.

El modelo ha sido implementado en la herramienta SLIM, que facilita la toma de decisiones por el usuario a través de una interfaz gráfica amigable y la comparación de diversas alternativas. El modelo propone como resultado gráfico de la estimación un valor de un único punto, sabiendo que se trata de la solución del 50% de probabilidad de una distribución normal. Es decir, se tiene una probabilidad del 50% de que el proyecto tenga una duración o un esfuerzo menor que la propuesta en la solución.

Viendo el vaso medio vacío, también existe la misma probabilidad de que el proyecto tenga una duración mayor o que emplee más esfuerzo. Por este motivo, la herramienta proporciona los valores estimados para todo el rango de probabilidades, de 1 a 99%, permitiendo al usuario elegir el riesgo con el que desea trabajar.

Tabla 3: SLIM: Distribución de probabilidad de que la duración sea menor que la indicada

Assurance Level (%)	Life Duration (Months)
1	8.10
5	8.91
10	9.34
15	9.63
20	9.87
25	10.06
30	10.24
35	10.41
40	10.56
45	10.72
50	10.86
55	11.01
60	11.16
65	11.32
70	11.49
75	11.66
80	11.86
85	12.10
90	12.39
95	12.82
99	13.63

Referencias bibliográficas

- [1] Putnam, L. H., Myers, W. "Industrial Strength Software". IEEE Computer Society Press. Los Alamitos, CA. 1997.
- [2] Putnam, L. H., Myers, W. "Measures for Excellence". Prentice Hall - Youdon Press. New Jersey. 1992.
- [3] Boehm, B. W. "Software Engineering Economics". Prentice Hall. New Jersey. 1981.
- [4] IFPUG. "Guidelines to Software Measurement". IFPUG (International Function Point Users' Group). <http://www.ifpug.org>. Westerville, OH. 1996.
- [5] IFPUG. "IFPUG Function Point Counting Practices: Manual Release 4.1". IFPUG (International Function Point Users' Group), <http://www.ifpug.org>. Westerville, OH. 2000. Traducido al castellano por AEMES. <http://www.aemes.org>. Madrid. 2002.
- [6] Boehm, R. "Function Point FAQ. Frequently Asked Questions (and Answers) Regarding Function Point Analysis". <http://ourworld.compuserve.com/homepages/softcomp/fpfaq.htm>. Software Composition Technologies, Inc. New Jersey. 1997.
- [7] Total Metrics, "Total Metrics Levels of Counting". <http://www.totalmetrics.com>. Victoria (Australia). 2001.
- [8] COCOMO Research Group. "COCOMO II Model Definition Manual" 1997. <http://sunset.usc.edu/COCOMOII/cocomo.html>.
- [9] Boehm B., Clark B., Horowitz E., Madachy R., Shelby R. y Westland C. "Cost Models for Future Software Life Cycle Processes: COCOMO 2.0". *Annals of Software Engineering*. Vol. 1 pp. 57-94. Amsterdam. 1995.

Esta web ha sido atacada



Autor: Luisa Morales Gómez-Tejedor Directora del Centro de Testing de Sopra PROFit Vocal del Comité del Software de la AEC.

Publicado en la Revista Calidad noviembre de 2008.

Con la frase “Esta web ha sido atacada” en inglés, se encontraron con estupor los clientes de una compañía californiana de modelismo de trenes cuya venta de un amplio surtido de maquetas no sólo se lleva a cabo a través de un gran almacén, sino también, y desde hace más de diez años, desde su página web. ¡Escalofriante! Media página web cubierta con la imagen de un hacker y el mensaje “Hacked by xxxx” seguido de una reivindicación islamista.

La revista Information Week explicaba en su número de febrero de 2007, en un artículo titulado “The hackers economy”, el funcionamiento de un nuevo mercado negro en el que se vende casi de todo.

Algunos ejemplos sacados de Trend Micro son: programas troyanos para robar información de cuentas, permisos de conducir, certificados de nacimiento, tarjetas de la Seguridad Social, números de tarjetas de crédito con su código de seguridad y fecha de caducidad, etc.

Más cerca de nosotros, geográficamente hablando porque la red nos ha convertido a todos en globales y en muchísimo más cercanos de lo que jamás hubiéramos podido imaginar, el diario El País publicaba el 11 de junio del pasado año que los datos de 120.000 usuarios españoles estaban en manos de “ciberpiratas”. Los hackers habían obtenido claves personales y bancarias de los clientes de una importante empresa española de Internet, tras haber atacado su base de datos. Además de obtener ilícitamente esos datos, los hackers lograron inyectar un virus en miles de páginas web pertenecientes a los usuarios afectados, entre los que figuraban conocidas empresas españolas.

Con el tiempo hemos aprendido a fortificar nuestros activos lógicos mediante antivirus, encriptaciones, Firewalls, etc., pero a menudo hemos dejado escapar algo que, de tan evidente, nos ha pasado sigilosamente desapercibido. ¿Hemos fortificado nuestras aplicaciones o nuestros sistemas informáticos? Como hemos podido ver, las aplicaciones son de alto valor para los hackers porque a través de ellas se manejan datos de clientes, tarjetas de crédito, identidades, seguridad social, impuestos, salud, etc., y todo ello es susceptible de ser sometido a fraude o hurto mediante la correspondiente e ilícita manipulación.

Las aplicaciones web son el objetivo número uno de los 'hackers'

El Informe sobre Amenazas de Symantec de 2006 sostenía que más del 90 por ciento de los ataques se habían producido en la capa de aplicaciones. Esta situación está igualmente refrendada por analistas como Gartner, cuyos análisis sustentan que el 75 por ciento de los ataques se produce en la capa de aplicaciones, y por el Wall Street Journal, que publicaba que el 69 por ciento de las vulnerabilidades documentadas en el año 2006 se habían producido en aplicaciones web.

El problema es, si cabe, más acuciante cuando se descubre que las aplicaciones web no sólo son el objetivo número uno de los hackers, sino que además la mayoría de los sitios web son vulnerables. Según la compañía Watchfire, el 90 por ciento de los sitios son vulnerables a ataques a sus aplicaciones.

Asimismo, Symantec afirma que el 78 por ciento de las vulnerabilidades son fácilmente explotables en las aplicaciones web. Según Gartner, en su informe Now is the time for security at application level (Ahora es el momento para la seguridad en el nivel de las aplicaciones) de diciembre de 2005, para el próximo año 2009, el 80 por ciento de las compañías habrá sufrido algún incidente de seguridad en las aplicaciones y, como resultado, reaccionarán creando roles en las organizaciones de desarrollo y de testing para asegurar que la seguridad se gestiona en el nivel de la aplicación (0,7 probabilidad).

Existen hoy en día más de doscientos tipos de amenazas catalogadas. A modo de ejemplo, se pueden mencionar algunas como Cookie poisoning, cuyo impacto es obtener el pirateo de la sesión con un posible resultado de robo de cualquier tipo de los ya mencionados en el presente artículo; Hidden fields, que produce una alteración del sitio web y provoca una transacción ilegal; Debug options, que puede conseguir un acceso no autorizado, comprometer la privacidad, así como el sitio en su conjunto; Cross Site scripting, que puede producir como consecuencia un robo de identidad; Stealth Commanding, que puede tener como consecuencia un acceso a información no pública; y SQL Injection, cuyo impacto es el acceso de lectura y escritura a bases de datos.

Si analizamos las consecuencias reales que pueden llegar a tener éstos y otros ataques, llegamos a la conclusión de que la seguridad de las aplicaciones es, sobre todo y ante todo, una cuestión de negocio, no únicamente un problema informático.

El foco de la solución

El contexto actual del desarrollo de aplicaciones web pasa por buscar la máxima calidad con la mínima inversión. Se valoran aspectos como el tiempo para estar en el mercado, la funcionalidad, facilidad de uso, rendimiento, disponibilidad, etc., pero en muchas ocasiones se pasan de puntillas otros aspectos del mismo orden de importancia, como los relacionados con la calidad interna del código en el que están escritas las aplicaciones, calidad que permite que una aplicación sea segura frente a amenazas, consistente en el tiempo, fácil de mantener y, por tanto, menos costosa para la organización. La solución, pues, apunta hacia la incorporación de la seguridad, como un componente más de la calidad, en el ciclo de desarrollo de software.

En la fase de requisitos se habrán recopilado los casos de abuso y se habrán convertido en requisitos de seguridad.

Los equipos de desarrollo diseñan las aplicaciones partiendo de requisitos funcionales y técnicos. La metodología, normas de acceso a bases de datos, reutilización de código y normas de codificación segura son, por tanto, componentes críticos de la definición técnica de los

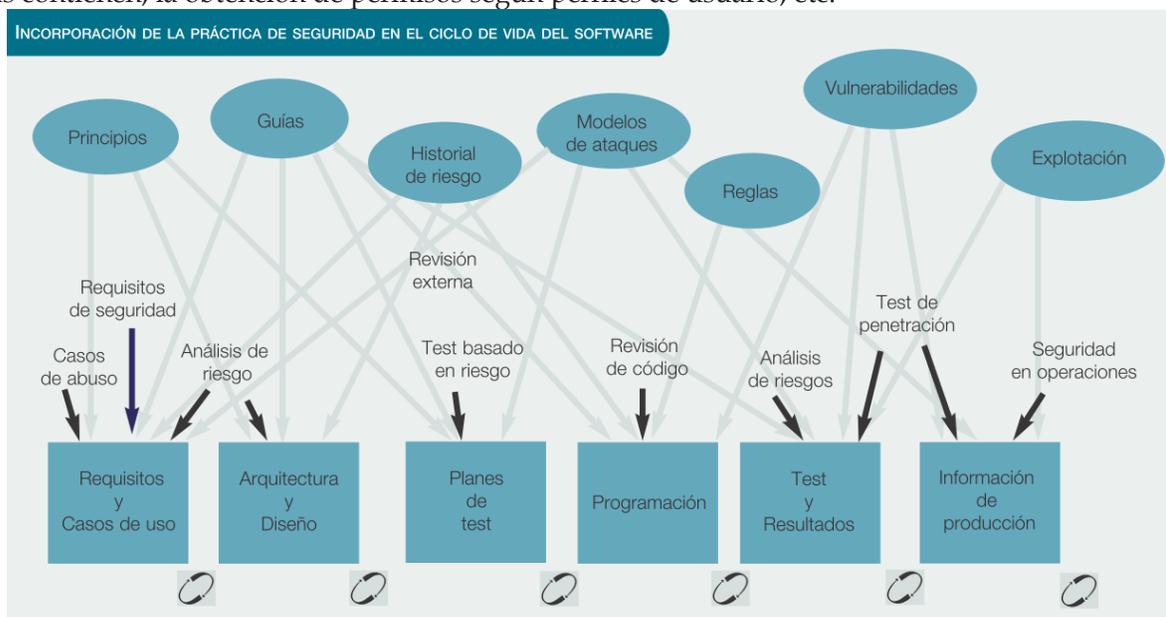
requisitos, que serán el punto de partida de la concepción y diseño del futuro sistema informático. Y desde ese punto inicial hasta la puesta en producción del sistema, en cada fase se deberán llevar a cabo pruebas orientadas a la verificación y validación de los mismos.

Las pruebas pueden llevarse a cabo de dos formas muy distintas, tanto por las características de su ejecución como por el momento en el que se abordan:

- Pruebas dinámicas que se llevan a cabo mediante la ejecución de la aplicación y sometiéndola a diferentes casos de comprobación, que puedan ayudar a aflorar errores ocultos en la misma.
- Pruebas estáticas para las que no es necesario que esté finalizada la aplicación, por lo que pueden abordarse desde que se escribe la primera línea de código, y consisten en el análisis minucioso de la calidad y seguridad de dicho código.

Pruebas dinámicas

Son pruebas orientadas a verificar la seguridad informática perimetral de un sistema y evitar así cualquier tipo de intrusión. Mediante estas pruebas se simula el comportamiento de usuarios maliciosos, orientado a conseguir cualquier información valiosa de los servidores de internet. Muchos problemas de seguridad pueden deberse a una mala configuración del software instalado, o a una inadecuada actualización del mismo. Por ello, las pruebas dinámicas van orientadas a detectar vulnerabilidades tanto de la propia aplicación en funcionamiento, como de todos los sistemas informáticos accesibles desde internet. Algunos ejemplos de verificación mediante este tipo de pruebas son los accesos a bases de datos, los servicios instalados, la robustez de las contraseñas, el posible robo o ataque de los archivos que las contienen, la obtención de permisos según perfiles de usuario, etc.



Gary McGraw: Software security building security in. Addison Wesley

Una vez detectadas las vulnerabilidades se establece una clasificación de los riesgos en base al impacto potencial de cada vulnerabilidad de seguridad, junto con la probabilidad de explotación de la misma. El impacto se valora en base a las consecuencias de una pérdida de la confidencialidad, integridad o disponibilidad de la información. La probabilidad se valora en base a los controles realmente existentes.

Pruebas estáticas

Es un hecho que, en más ocasiones de las deseadas, los equipos de desarrollo llevan a cabo la construcción del código sin tener en cuenta criterios de seguridad. Esta programación “no segura” puede dar lugar a que un “atacante” se salte las protecciones del sistema y pueda acceder a datos e información privada.

Las acciones de verificación mediante este tipo de pruebas van orientadas a evitar el acceso ilegal a las cuentas y datos de los usuarios, acceso a áreas privadas, suplantación de usuarios, puertas traseras, etc. Son las pruebas de caja blanca las que analizan la seguridad del código a través del análisis del código fuente y de las dependencias existentes entre los diferentes módulos, confrontándolo con reglas de codificación segura para, finalmente, proporcionar una lista de vulnerabilidades detectadas, categorizadas en función de criterios de seguridad previamente definidos.

Su propia condición de pruebas de caja blanca, que analizan el código contrastándolo con reglas previamente establecidas, hace que este tipo de comprobaciones puedan formar parte de un grupo más extenso de pruebas, igualmente orientadas al análisis del código.



El problema es, si cabe, más acuciante cuando se descubre que las aplicaciones web no sólo son el objetivo número uno de los *hackers*, sino que además la mayoría de los sitios web son vulnerables

En este sentido, varias empresas españolas han creado equipos dedicados al análisis global del código fuente de las aplicaciones, categorizando los distintos análisis según el objetivo perseguido:

- **Mantenibilidad.** Definida como la facilidad de un producto software para ser modificado de manera eficiente por la inclusión de correcciones, mejoras o adaptación del software a cambios en el entorno, en los requerimientos o en las especificaciones funcionales. Es el atributo de calidad del software que más directamente influye en los costes y necesidades del mantenimiento. Las métricas que se utilizan para medir la mantenibilidad de un sistema pueden variar según el lenguaje de programación. A modo de ejemplo, se pueden mencionar para COBOL: sencillez, autodescripción, modularidad, legibilidad, modificabilidad y cumplimiento de estándares; y para JAVA: optimización, legibilidad, sencillez, complejidad, estandarización y acoplamiento.
- **Rendimiento.** Definido como la capacidad de un sistema para ofrecer la funcionalidad prevista con las prestaciones requeridas y el menor consumo posible de recursos. El rendimiento puede analizar tanto el lenguaje de programación como las bases de datos y los lenguajes de ejecución de trabajos.
- **Seguridad.** Definida como la disposición de un sistema para mantener la integridad de acceso a datos sensibles desde fuentes no autorizadas. Habitualmente, las vulnerabilidades que se detectan se clasifican, según su impacto, en tres niveles: graves, avisos e informativos.

El análisis de seguridad y calidad de código proporciona una ayuda muy valiosa en tres aspectos:

- Información interna sobre la calidad y seguridad del software. Proporciona información en tres niveles distintos: la calidad y seguridad del código que se pasa a producción, el grado de calidad y seguridad de la instalación y la evolución de la

- Año 2005 (con pruebas estáticas):
 - Número de errores detectados en las pruebas estáticas: 295.
 - Número de errores detectados en las pruebas unitarias: 269.
 - Número de errores detectados en las pruebas de sistemas: 569.
 - Número de errores detectados en las pruebas de aceptación: 192.
 - Coste de las pruebas de aceptación: 220.000 libras.

Los errores que se encuentran en etapas tempranas del proyecto se arreglan con un bajo coste y se evita el efecto multiplicador hacia las etapas posteriores. O, dicho de otra manera, cuanto más tarde se detecta un error, más alto es el coste de su solución ya que afecta a un mayor número de componentes (módulos, ficheros, bases de datos, etc.) que deben ser analizados y corregidos. Por tanto, una gran parte de los errores que se cometen en el desarrollo de los sistemas se pueden descubrir pronto o al tiempo en que éstos se escriben a través del testing estático. Errores que pueden afectar negativamente a la seguridad, rendimiento y mantenibilidad de los sistemas.

Las pruebas estáticas son una práctica efectiva y de alto retorno de valor que deben llevarse a cabo a lo largo de todo el ciclo de vida de desarrollo, haciendo un mayor énfasis durante la fase de codificación. Esto ayudará, significativamente, a que los errores más importantes se encuentren antes. La clave está en prevenir, no actuar reactivamente.

Casos prácticos

Los profesionales de la calidad del software en España podíamos decir que no somos tan valorados como en otros países; la tremenda demanda del mantenimiento de software que existe en España hace que la calidad no se primordial para muchos directivos y por consiguiente para sus empresas.

Por ello ha sido siempre una máxima de este comité el poder escribir sobre soluciones reales que muestren que la llamada “teoría” se puede aplicar con éxito en las empresas.

Aquí hablamos del ámbito sanitario y del uso de la calidad del software en los Juegos Olímpicos.

Artículos presentados en esta selección:

- [La certificación CMMI-DEV, nivel 3 de unos Juegos Olímpicos](#)
- [Procesos y aspectos clave en el sector de los productos sanitarios](#)



personas



procesos



tecnologías

La certificación CMMI-DEV, nivel 3 de unos Juegos Olímpicos

GESTIÓN POR PROCESOS / ARTÍCULO

La certificación CMMI-DEV, nivel 3 de unos Juegos Olímpicos



Laura Costa,
Software Dev., CMMI & Quality
Process Manager de Atos



Domingo Gaitero,
Business Director de Atos
Consulting, Vicepresidente
del Comité CSTIC
de la AEC



Ana González,
Manager de Atos Consulting,
Vocal del Comité CSTIC
de la AEC



Álvaro Pérez,
Sochi 2014 Integration
Manager de Atos

Cada dos años, alternando invierno y verano, los Juegos Olímpicos tienen lugar en diferentes ciudades, con diferentes personas, diferentes tecnologías, diferentes retos... y siempre con una fecha límite inamovible.



10 | CALIDAD julio-septiembre 2011

Autores: Laura Costa, Software Dev., CMMI & Quality Process Manager de Atos, Domingo Gaitero, Executive Director de Atos Consulting, Vicepresidente del Comité CSTIC de la AEC, Ana González, Manager de Atos Consulting, Vocal del Comité CSTIC de la AEC, Álvaro Pérez, Sochi 2014 Integration Manager de Atos

Publicado en la revista Calidad - N° III • 2011

Cada dos años, alternando invierno y verano, los Juegos Olímpicos tienen lugar en diferentes ciudades, con diferentes personas, diferentes tecnologías, diferentes retos... y siempre con una fecha límite inamovible.

Atos, Socio Tecnológico Oficial del Comité Olímpico Internacional (IOC Worldwide

Information Technology Partner), es la encargada de crear y asegurar la compleja infraestructura y los sistemas de TI que proporcionan la logística y el envío de resultados e información deportiva a los medios de comunicación acreditados.

El pasado 23 de junio de 2010, la unidad Major Events (MEV) de Atos, encargada de gestionar y asegurar el éxito de este acontecimiento mundial, se acreditó en el nivel 3 de madurez del modelo de mejora de procesos CMMI-DEV (Capability Maturity Model Integration for Development).

A continuación, se presentan los elementos clave que determinaron que MEV iniciara un programa de mejora de procesos, a consecuencia del cual se obtuvo la certificación, y los principales retos a los que tuvo que hacer frente el equipo de Atos Consulting, colaborador en el programa.

¿Para qué necesitan una certificación?

MEV se inició como proveedor del software de los Juegos Olímpicos en Barcelona 92, y desde Salt Lake City 2002 es el encargado de desarrollar, integrar y garantizar el correcto funcionamiento de los complejos sistemas de información de los Juegos Olímpicos. Este contrato, a largo plazo con el IOC, está basado en una relación de confianza y en un rendimiento probado donde, a priori, no es necesaria una certificación para demostrar unos buenos resultados. El único requisito exigido es eficiencia y garantía de fallos cero.

Para satisfacer con garantía este requisito es necesario contar con un equipo de personas altamente cualificadas, con la institucionalización de unos procesos robustos y con una alta

eficiencia y disponibilidad de la tecnología que soporte estos procesos.

Personas, procesos y tecnología son las piezas básicas para conseguir los objetivos de negocio de cualquier organización.

En el caso que nos ocupa estos objetivos se concretan en los siguientes:

1. Ofrecer mayor calidad.
2. Mejorar la eficiencia.
3. Reducir costes.
4. Aumentar la satisfacción del cliente.

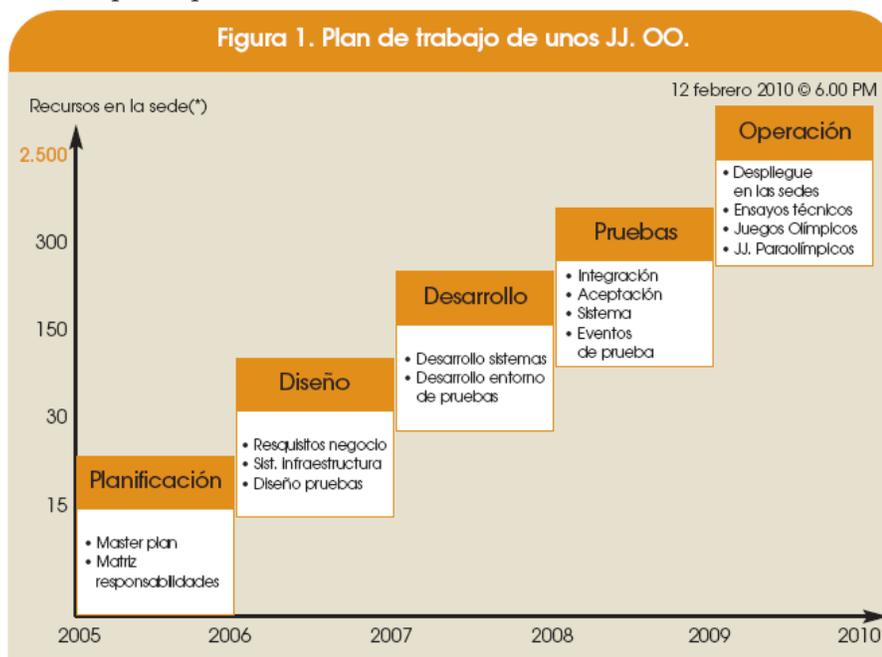
La identificación de estos objetivos viene condicionada, en parte, por factores externos a la organización relacionados con el complejo entorno globalizado al que se están orientando las tecnologías de la información.

Este entorno presenta oportunidades y amenazas para el negocio. Oportunidades como la posibilidad de realizar *off-shoring* de determinados procesos, aumentando el nivel de especialización de los equipos. Y amenazas como la necesidad de mantener la competitividad y la cuota de mercado ante la competencia, sobre todo en costes, de las empresas asiáticas.

Implícitamente en todos estos factores se halla el compromiso de la organización de conseguir la sostenibilidad de los procesos, con independencia de las personas que los ejecuten o la tecnología que los soporte.

En resumen, para hacer frente a las oportunidades y amenazas del mercado, y alcanzar los objetivos de negocio, se identificaron las siguientes acciones para el programa de mejora:

- Evolución del modelo de procesos hacia un enfoque industrializado, con equipos de trabajo cada vez más especializados y la participación de centros de *off-shoring* que permitiesen mejorar la eficiencia de los procesos. Esta especialización requería también un cambio organizacional, lo que suponía transformar una estructura vertical orientada a proyectos en una estructura transversal orientada a productos que evolucionasen en cada proyecto, permitiendo aprovechar sinergias entre ellos.
- Identificación del propietario de proceso. Al establecer una estructura transversal, lo

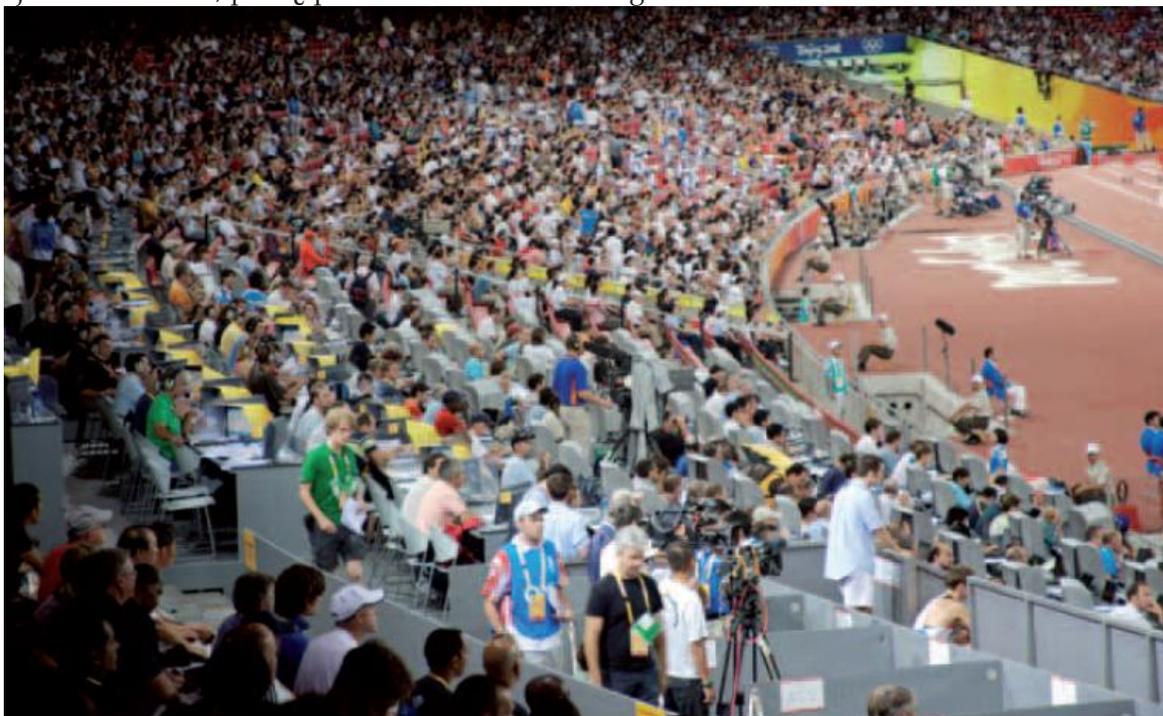


común es que las actividades de un proceso estén distribuidas entre diferentes áreas funcionales, con el riesgo de que nadie se responsabilice del mismo ni de sus resultados finales. La figura del propietario del proceso es la de una persona con autoridad y conocimiento del proceso, encargada de promover su definición y mejora, comprobar que

se utilice y evaluar los cambios que considere adecuados. Como elemento coordinador entre los distintos propietarios de procesos se estableció un equipo de mejora de procesos.

- Actualización de la tecnología. En consonancia con la iniciativa de Atos para la estandarización y centralización de las herramientas de soporte al desarrollo (Shared Services Center), fue necesario renovar las herramientas que soportaban los procesos. El propietario del proceso fue el encargado de crear un equipo de trabajo para realizar el análisis coste-beneficio del uso de las herramientas propuestas por la organización y evaluar las necesidades de adaptación al nuevo modelo de procesos. La alta dependencia de las herramientas en los proyectos olímpicos, en cuanto a rendimiento, disponibilidad, seguridad, etc., hicieron que éste fuese uno de los aspectos más críticos del programa de mejora.
- Refinamiento de métricas para evaluar y controlar que las acciones de mejora estaban sirviendo para alcanzar los objetivos definidos. El refinamiento estaba orientado tanto a controlar el grado de implantación del programa de mejora como a evaluar el rendimiento de cada uno de los proyectos bajo el nuevo modelo de procesos, tomando como referencia las métricas establecidas por el cliente (IOC) en cada contrato.

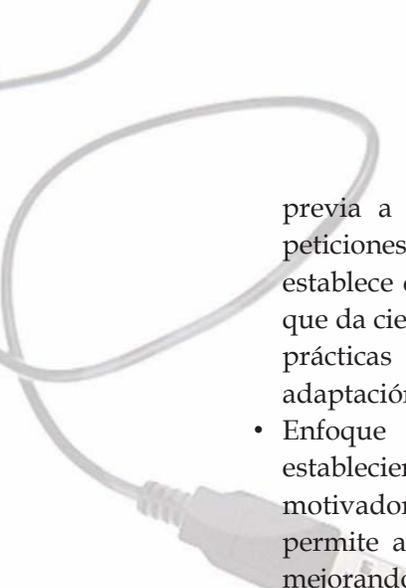
Estas oportunidades de mejora fueron el detonante principal para iniciar un Programa de Mejora de Procesos, pero ¿qué estándar o modelo elegir?



Elección de CMMI como modelo de mejora de procesos

Las razones principales para decantarse por CMMI como modelo para la mejora de procesos fueron las siguientes:

- Flexibilidad de CMMI para integrar prácticas específicas del negocio olímpico no contempladas en las áreas de proceso propias del modelo; por ejemplo, el proceso de gestión del conocimiento, en el que se establece una fase de transferencia de conocimiento



previa a la toma de requisitos con el cliente/usuario que minimiza el número de peticiones de cambio posteriores. El valor de CMMI, como modelo de mejora, es que establece qué es lo que hay que hacer para mejorar, pero no indica cómo hacerlo, por lo que da cierto grado de flexibilidad a la organización para poder implementar y adaptar las prácticas a la casuística de su negocio o de sus proyectos, siempre y cuando esta adaptación permita cumplir los objetivos establecidos en cada área de proceso de CMMI.

- Enfoque del proceso de evaluación como mecanismo para sostener la mejora, estableciendo puntos de control a lo largo del programa que proporcionan un efecto motivador y una referencia objetiva sobre la implantación de la mejora. La evaluación permite a la organización pasar de la creencia al hecho, del “pensamos que estamos mejorando” al “sabemos que estamos mejorando”.

El proceso de evaluación se realizó de forma continua a lo largo de todo el programa de mejora; para ello se contó con la experiencia y la visión objetiva de un Lead Appraiser autorizado por el SEI (Software Engineering Institute). Esto permitió focalizar y priorizar las oportunidades de mejora y facilitó la interpretación de las prácticas del CMMI con respecto al modelo de negocio de MEV.

Retos durante el programa de mejora y la certificación

Los retos típicos de cualquier programa de mejora hacen referencia, principalmente, a la resistencia al cambio de las personas, al uso de tecnología no adaptada al proceso o la implantación de procesos que no reflejan el modo de trabajo de la organización. El caso que nos ocupa no fue diferente en este sentido, pero además se presentaron otros desafíos:

- Modelo de negocio olímpico. La duración de un proyecto olímpico es de cuatro años. Si tenemos en cuenta que cada dos años se celebran unos Juegos Olímpicos, significa que en un momento dado el mismo equipo está trabajando simultáneamente, al menos, para dos JJ. OO. (el proyecto de Londres 2012 se inició en 2008, en ese periodo de tiempo se habrán celebrado los Juegos de Beijing 2008 y Vancouver 2010).
- Elección de proyectos para la evaluación. Una de las principales dificultades fue la elección de los proyectos que participarían en la evaluación. Debido a su esquema de trabajo con ciclos de cuatro años, cuando se decidió iniciar el proceso de evaluación ningún proyecto disponía de evidencias para cubrir el alcance de todas las áreas de proceso del modelo CMMI; Beijing acababa de terminar, Vancouver estaba en fase de despliegue y en Londres había procesos que aún no se habían ejecutado. Además, tanto Beijing como Vancouver se habían realizado con el modelo de procesos anterior. Finalmente, se decidió seleccionar Londres como proyecto focus de la evaluación, y Beijing y Vancouver como proyectos non-focus que diesen cobertura a aquellas áreas de proceso en las que Londres aún no tenía evidencias.
- Agenda del equipo de MEV. Otro de los retos fue encajar las reuniones de seguimiento y las sesiones de formación/divulgación del programa de mejora en la ajustada agenda del equipo de MEV, que tiene su día a día planificado a cuatro años vista, incluyendo visitas periódicas a la sede olímpica, sede a la que finalmente se desplazan cinco meses, dos meses antes de los JJ. OO., la celebración de los JJ. OO. y dos meses después. Para poder disponer siempre de un interlocutor del programa de mejora se constituyó un equipo de mejora compuesto por propietarios de proceso con el mismo grado de implicación en la mejora.
- Personal altamente cualificado. Sin duda, el mayor reto al que se enfrentó el equipo de

Atos Consulting, y también el más gratificante, fue colaborar con unos profesionales altamente cualificados y con gran experiencia, tanto en la definición de procesos como en su despliegue, que exigía un gran pragmatismo y efectividad a la hora de enfocar la definición e implantación de los procesos.

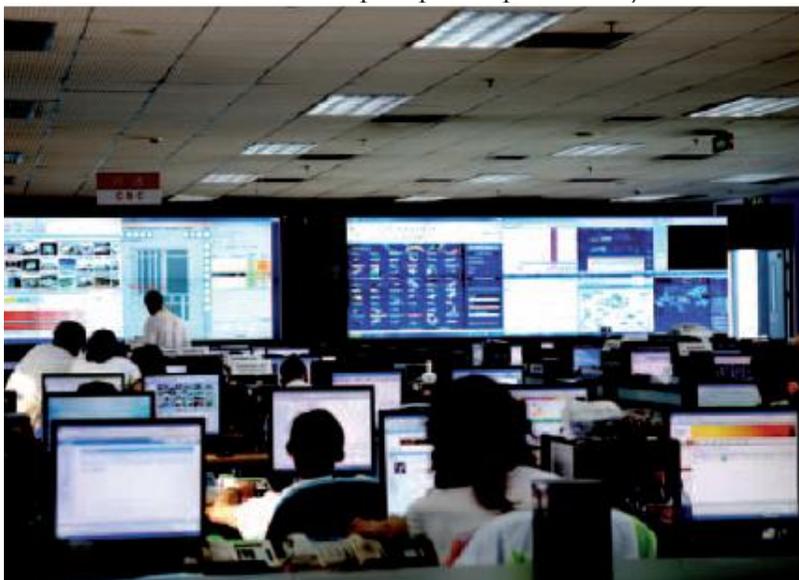
Factores clave del éxito

El éxito del Programa de Mejora de Procesos es haber conseguido la certificación CMMI nivel 3 de madurez como consecuencia de un enfoque mucho más amplio en el que la certificación es un componente más de la mejora continua y no un fin en sí mismo.

De este modo, se asegura que la organización inicia un camino seguro hacia la mejora, garantizando la sostenibilidad en el tiempo de los buenos resultados.

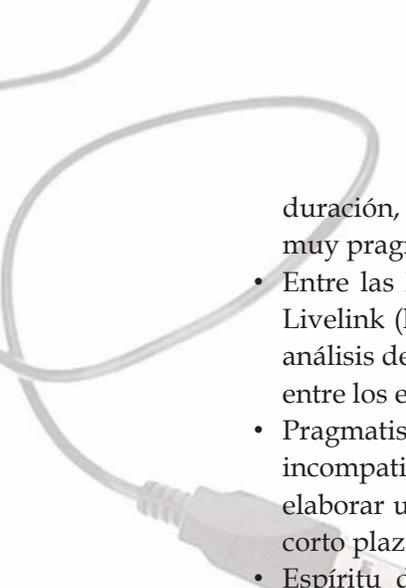
Otros de los factores clave que contribuyeron al éxito de este programa fueron:

- Liderazgo de la dirección. Para que un programa de mejora continua tenga éxito, es importante que éste se inicie al más alto nivel de la organización. Sólo con el apoyo incondicional de la dirección se producirá la aportación de recursos y la asignación de responsabilidad y autoridad a los líderes de la mejora para que éstos cumplan su cometido.
- En nuestro caso, el compromiso e implicación de la dirección se contagió al resto de la organización y fue capaz de transmitir a todos los miembros de MEV la convicción de que todos ellos tenían ideas que aportar para la mejora continua.



Unos procesos arraigados en la cultura y la forma de trabajar de la organización, unas personas motivadas y unas herramientas adaptadas a su forma de trabajo son, en resumen, los factores de éxito de la certificación CMMI nivel 3 de madurez de Major Events

- Uno de los efectos del apoyo exitoso de la dirección fue sin duda la convicción de los miembros de la organización en que la mejora de procesos era posible.
- Gestión del conocimiento. Uno de los mecanismos utilizados para conseguir la implicación y participación de todos en la mejora continua fue contar con herramientas que apoyasen la gestión del conocimiento. Estas herramientas buscaban transferir el conocimiento y la experiencia del grupo para que pudiese ser utilizado por otros miembros.
- En el negocio de MEV, la gestión de conocimiento es uno de sus procesos clave, ya que sus proyectos empiezan con equipos pequeños en cada país y, poco a poco, van incorporando recursos, sin experiencia en el negocio y a veces en proyectos/actividades de poca



duración, por lo que la curva de aprendizaje debe ser corta y el enfoque de la formación muy pragmático.

- Entre las herramientas con las que cuentan destacan: librería de activos de procesos en Livelink (herramienta corporativa de gestión documental), wiki, blogs, FAQ, prácticas de análisis de lecciones aprendidas tras cada proyecto, fases de transferencia de conocimiento entre los equipos de los distintos JJ. OO., etc.
- Pragmatismo. El alto grado de pragmatismo exigido en los procesos no resultó incompatible con su alto grado de formalización. La fase de definición se orientó a elaborar unos procesos y procedimientos prácticos, eficaces y con resultados evaluables a corto plazo.
- Espíritu deportivo del equipo MEV. Cuando se inicia un proceso de mejora en una organización, lo habitual es obtener una respuesta de resistencia al cambio por parte de sus miembros. En el caso de MEV, ocurrió lo contrario, encararon el proceso de cambio con espíritu deportivo, por cuanto lleva implícito de afán de superación y de mejora día a día.
- La ambición por ser mejores, junto con la confianza que da pertenecer a un equipo que comparte un objetivo común y una larga trayectoria de éxito, hizo que para ellos cualquier obstáculo se convirtiese en una motivación.
- Este efecto se acrecienta si se tiene en cuenta el orgullo y la responsabilidad que supone realizar un trabajo cuyo resultado final hace sentir a sus participantes parte de un evento que llega a cientos de millones de personas.

Unos procesos arraigados en la cultura y la forma de trabajar de la organización, unas personas motivadas y unas herramientas adaptadas a su forma de trabajo son, en resumen, los factores de éxito de la certificación CMMI nivel 3 de madurez de Major Events.

El Programa de Mejora de Procesos estuvo liderado por Llorenç Àguila, como responsable del equipo Global Processes de MEV, en colaboración con personal de Atos Consulting de la práctica de IT Governance, dirigida por Domingo Gaitero, y con José Arias como Lead Appraiser del European Software Institute (ESI) y representante del SEI (Software Engineering Institute) en España.

Procesos y aspectos clave en el sector de los productos sanitarios

Q GESTIÓN POR PROCESOS / ARTÍCULO

Procesos y aspectos clave en el sector de los productos sanitarios



José Alberto González,
Ingeniero de Calidad GMV
Vocal del Comité de CSTIC
de la AEC

Introducción

Los fabricantes de productos sanitarios están obligados a conocer en detalle las regulaciones sanitarias vigentes en los mercados a los que desean enfocarse. Con independencia del nivel de exigencia y de la armonización asociada a dichas regulaciones, existen unas normas genéricas, armonizadas internacionalmente, que establecen los requisitos esenciales que debe satisfacer un fabricante respecto a su organización y a sus instalaciones, y respecto a la documentación técnica y clínica del producto para poder introducirlo en el entorno sanitario.

El presente artículo se enfoca en estas normas genéricas armonizadas internacionalmente, las cuales constituyen un marco normativo globalmente aceptado en el sector de los productos sanitarios (ver figura 1). Estas normas describen, en definitiva, los procesos y aspectos clave que deben preservar los fabricantes de productos sanitarios. La figura 2 enmarca estos procesos y aspectos clave en el ciclo de vida del producto sanitario, considerando tanto la fase de producción como la de postproducción.

Este conocimiento ha sido adquirido y puesto en práctica por GMV durante el desarrollo y la producción de su primer producto sanitario en el mercado europeo: radiance, un sofisticado sistema de planificación para radioterapia intraoperatoria, el cual está llamado a fomentar la estandarización, la eficacia y el uso creciente de esta disciplina.

1. Sistema de gestión de la calidad

1.1. Introducción

La norma ISO 13485 especifica los requisitos para un sistema de gestión de la calidad para fabricantes de productos sanitarios. Es una norma basada en un enfoque a procesos para la gestión de la calidad, elaborada sobre ISO 9001, y cuyo propósito es dar cobertura a los requisitos regulatorios. Por tanto, suele ser la referencia a la que también acuden los organismos de certificación para evaluar la capacidad de un fabricante para cumplir, además de los de sus clientes, los requisitos regulatorios.

1.2. Requisitos regulatorios

La norma señala que los fabricantes deben ser estrictos con las regulaciones sanitarias relevantes en sus mercados, y que deben mantenerse alerta frente a posibles cambios o a nuevos requisitos regulatorios para poder así reaccionar a ellos a las fuentes de aplicación. Cualquier requisito regulatorio aplicado o no aplicable debe estar identificado y adecuadamente justificado en el manual de la calidad.

El fabricante debe definir el periodo de vida del producto sanitario y, en función de éste y de los requisitos regulatorios relativos a este punto, establecer unos tiempos mínimos de retención de los documentos y de los registros asociados al producto.

Las regulaciones sanitarias requieren que el fabricante designe a una persona específica, con determinadas responsabilidades, como responsable del producto sanitario de cara al seguimiento del mismo en la etapa de postproducción y al informe de incidentes derivados del uso del producto.

En función de la naturaleza y de las características inherentes de un producto sanitario, puede ser necesario el establecimiento de medidas de control sobre el entorno de trabajo y sobre el personal involucrado para asegurar que la calidad del producto está siendo preservada durante la fabricación. Este aspecto se encuentra más desarrollado en la norma ISO 13485 frente a ISO 9001. Tal es así, que las regulaciones sanitarias requieren la inspección de las dependencias del fabricante implicadas en la fabricación del producto sanitario.

1.3. Expediente técnico del producto sanitario

Para cada tipo o modelo de producto sanitario, el fabricante debe preparar y mantener un expediente técnico que contenga o

Autor: José Alberto González, Ingeniero de Calidad GMV Vocal del Comité de CSTIC de la AEC.

Publicado en la revista Calidad - Nº III • 2011

Los fabricantes de productos sanitarios están obligados a conocer en detalle las regulaciones sanitarias vigentes en los mercados en los que desean enfocarse. Con independencia del nivel de exigencia y de burocracia asociado a dichas regulaciones, existen unas normas genéricas, armonizadas internacionalmente, que establecen los requisitos esenciales que debe satisfacer un fabricante respecto a su organización y a sus instalaciones, y respecto a la documentación técnica y clínica del producto para poder

introducirlo en el entorno sanitario.

El presente artículo se enfoca en estas normas genéricas armonizadas internacionalmente, las cuales constituyen un marco normativo globalmente aceptado en el sector de los productos sanitarios (ver figura 1). Estas normas describen, en definitiva, los procesos y aspectos clave que deben preservar los fabricantes de productos sanitarios. La figura 2 enmarca estos procesos y aspectos clave en el ciclo de vida del producto sanitario, considerando tanto la fase de producción como la de postproducción.

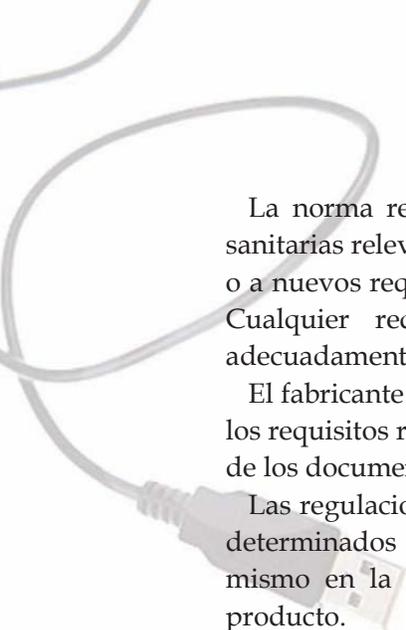
Este conocimiento ha sido adquirido y puesto en práctica por GMV durante el desarrollo y la producción de su primer producto sanitario en el mercado europeo: radiance, un sofisticado sistema de planificación para radioterapia intraoperatoria, el cual está llamado a fomentar la estandarización, la eficacia y el uso creciente de esta disciplina.

1. Sistema de gestión de la calidad

1.1. Introducción

La norma ISO 13485 especifica los requisitos para un sistema de gestión de la calidad para fabricantes de productos sanitarios. Es una norma basada en un enfoque a procesos para la gestión de la calidad, elaborada sobre ISO 9001, y cuyo propósito es dar cobertura a los requisitos regulatorios. Por tanto, suele ser la referencia a la que también acuden los organismos de certificación para evaluar la capacidad de un fabricante para cumplir, además de los de sus clientes, los requisitos regulatorios.

1.2. Requisitos regulatorios



La norma reitera que los fabricantes deben ser estrictos conocedores de las regulaciones sanitarias relevantes en sus mercados, y que deben mantenerse alerta frente a posibles cambios o a nuevos requisitos regulatorios para poder así reaccionar a ellos si les fueran de aplicación. Cualquier requisito regulatorio excluido o no aplicable debe estar identificado y adecuadamente justificado en el manual de la calidad.

El fabricante debe definir el periodo de vida del producto sanitario y, en función de éste y de los requisitos regulatorios relativos a este punto, establecer unos tiempos mínimos de retención de los documentos y de los registros asociados al producto.

Las regulaciones sanitarias requieren que el fabricante designe a una persona específica, con determinados requisitos, como responsable del producto sanitario de cara al seguimiento del mismo en la etapa de postproducción y al informe de incidentes derivados del uso del producto.

En función de la naturaleza y de las características inherentes de un producto sanitario, puede ser necesario el establecimiento de medidas de control sobre el entorno de trabajo y sobre el personal involucrado para asegurar que la calidad del producto esté siendo preservada durante la fabricación. Este aspecto se encuentra más desarrollado en la norma ISO 13485 frente a ISO 9001. Tal es así, que las regulaciones sanitarias requieren la inspección de las dependencias del fabricante implicadas en la fabricación del producto sanitario.

1.3. Expediente técnico del producto sanitario

Para cada tipo o modelo de producto sanitario, el fabricante debe preparar y mantener un expediente técnico que contenga o referencie todos los documentos de especificaciones del producto y los requisitos del sistema de gestión de la calidad. Estos documentos deben definir el proceso de fabricación completo y, si es aplicable, la instalación y el servicio postventa.

A efectos regulatorios, este expediente deberá ser complementado con los datos o informes de la evaluación o investigación clínica, el expediente de gestión de los riesgos, los detalles de acondicionamiento y etiquetado del producto, las instrucciones o manuales de uso y, también importante, la listas de comprobación de los requisitos regulatorios cumplimentadas por el propio fabricante, así como su declaración de conformidad.

La mayoría de las regulaciones sanitarias establecen una clasificación de los productos sanitarios en función de la gravedad de las lesiones o daños que su utilización puede ocasionar sobre las personas, normalmente de menor a mayor gravedad. El fabricante debe posicionar su producto sanitario, normalmente en el Expediente de Gestión de los Riesgos (ver sección 2.2), con respecto a esta clasificación. Esta clasificación determinará el grado de detalle con que las autoridades sanitarias, normalmente a través de organismos notificados, examinarán el Expediente Técnico del producto sanitario antes de otorgar cualquier certificación o marca, y autorizar así su puesta en el mercado.

1.4. Sistema de 'feedback'

El fabricante debe establecer y documentar un sistema de feedback de sus clientes y usuarios para la pronta detección de potenciales problemas durante el uso del producto y su inmediata introducción en los procesos de acciones correctivas y preventivas.

Cualquier reclamación debe ser registrada, debidamente investigada, y materializarse en una acción correctiva o preventiva.

Cualquier reclamación para la que el fabricante decida finalmente no adoptar ninguna acción debe estar debidamente justificada y autorizada. El fabricante debe conservar todos los

registros de las investigaciones que realice como resultado de una reclamación.



1.5. Sistema de vigilancia

El fabricante debe establecer y documentar un sistema de vigilancia para atender los potenciales incidentes derivados del uso del producto (ver figura 3). Por incidente se entiende cualquier lesión o daño experimentado por alguna persona que se sospecha o se sabe que está ocasionado por el uso del producto.

Estos potenciales incidentes pueden ser comunicados al fabricante directamente por sus clientes y usuarios o bien por las autoridades sanitarias ya alertadas; también pueden originarse por personal del fabricante que detecte alguna anomalía sobre un lote de fabricación que ya se hubiera distribuido.

El fabricante tiene la obligación de analizar el potencial incidente e informar, si lo considera tal, a la autoridad sanitaria correspondiente.

Ésta debe asimismo analizar el incidente, bien para descartarlo como tal, bien para confirmarlo y exigir al fabricante una investigación exhaustiva del mismo y la adopción de las medidas que se derivasen para mitigarlo.

En el transcurso de la mitigación del incidente, suele ser necesaria la comunicación con los clientes, normalmente por medio de notas de aviso, ante la adopción de una determinada acción correctiva de seguridad sobre un producto en campo, es decir, en uso en el mercado, o incluso ante la retirada del producto.

Estas notificaciones, conocidas en el sector como FSCA (Field Safety Corrective Action) y FSN (Field Safety Notice), están reguladas en formato y contenido por las autoridades sanitarias, por quienes deben ser también autorizadas, por lo que el fabricante debe adoptarlas en sus procesos de vigilancia.

2. Proceso de gestión de los riesgos

2.1. Introducción

La norma ISO 14971 proporciona a los fabricantes de productos sanitarios un marco dentro del cual aplicar sistemáticamente la experiencia, el entendimiento y el buen juicio en la gestión de los riesgos asociados al uso de los productos sanitarios. Estos riesgos pueden afectar principalmente a los pacientes, pero también a otras personas, por ejemplo, cuidadores, operadores, etc., así como a otros equipos y al entorno.



Según esta norma, el fabricante de un producto sanitario debe establecer, documentar y mantener a lo largo del ciclo de vida de su producto un proceso para identificar los peligros asociados al uso del producto, estimar y evaluar los riesgos asociados, controlar estos riesgos y monitorizar la efectividad de estos controles.

2.2. Expediente de gestión de los riesgos

El proceso de gestión de los riesgos asociados al uso de un producto sanitario debe materializarse en un Risk Management File o Expediente de Gestión de los Riesgos (ver figura 4), el cual debe incluir lo siguiente:

- La definición del proceso de gestión de los riesgos.
- El Plan de Gestión de los Riesgos (PGR).
- El registro (o base de datos) de riesgos.
- El Informe de Gestión de los Riesgos.
- La información de producción y postproducción.

2.2.1. Plan de gestión de los riesgos (PGR)

Al arrancar el ciclo de vida del producto, el fabricante debe elaborar un Plan de Gestión de los Riesgos, el cual debe incluir lo siguiente:

- El alcance de las actividades de gestión de los riesgos planificadas, identificando y describiendo el producto sanitario y las fases del ciclo de vida para las cuales cada elemento del plan es aplicable.
- La asignación de autoridades y responsabilidades.
- Los requisitos de revisión de las actividades.
- Los criterios para aceptabilidad de los riesgos, basados en la política definida por el fabricante.
- Las actividades de verificación.
- Las actividades relacionadas con la obtención y la revisión de la información relevante de las fases de producción y postproducción.

Si este plan cambia durante el ciclo de vida del producto, se debe mantener un registro de cambios.

2.2.2. Registro (o base de datos) de riesgos

El registro (o base de datos) de riesgos es el repositorio en el cual se almacenan los riesgos al tiempo que van siendo identificados y procesados a lo largo de todo el ciclo de vida del producto.

Para cada riesgo se mantendrá toda la información necesaria, incluyendo al menos los datos: identificador, título, descripción, fecha de detección, severidad, probabilidad, aceptabilidad,

análisis, estado y fecha y comentarios de revisión.

2.2.3. Informe de gestión de los riesgos

Previa liberación del producto sanitario ante cada distribución comercial, el fabricante debe llevar a cabo una revisión del proceso de gestión de los riesgos, cuyos resultados deben ser registrados en un Informe de Gestión de los Riesgos, el cual debe confirmar lo siguiente:

- El PGR ha sido seguido apropiadamente.
- El riesgo residual global es aceptable.
- Se están proporcionando los métodos apropiados para obtener la información relevante de las fases de producción y postproducción.

2.2.4. Información de producción y postproducción

La información recogida durante las fases de producción y postproducción del producto sanitario debe ser utilizada para realimentar el proceso de gestión de los riesgos. Dicha información puede contribuir a identificar nuevos riesgos o a hacer variar las estimaciones de

riesgos ya identificados, la evaluación de la aceptabilidad de los riesgos, las medidas de control de los riesgos, etc. Cuando así sea, dicha información debe conservarse como parte del Expediente de Gestión de los Riesgos.



2.3. Subprocesos de la gestión de los riesgos

El proceso de gestión de los riesgos debe estar claramente definido y documentado, formando parte del sistema de gestión de la calidad del fabricante, o del Expediente Técnico del producto sanitario en cuestión. La figura 5 representa un esquema básico de su aspecto. En esta sección se desgranarán sus etapas o subprocesos.

2.3.1. Análisis de los riesgos

Como primer paso en la gestión de los riesgos, se debe realizar un Análisis de los Riesgos para el producto sanitario en cuestión.

El fabricante debe documentar el uso previsto del producto, así como el, razonablemente previsible, mal uso. El fabricante debe identificar y documentar aquellas características cualitativas y cuantitativas que podrían afectar al uso seguro del producto y, si procede, definir sus límites.

El fabricante debe recopilar toda la documentación sobre los peligros conocidos o previsibles asociados con el producto sanitario, en condiciones normales y en condiciones de fallo.

Para cada peligro identificado, se debe estimar el riesgo asociado usando la probabilidad de

ocurrencia del peligro. Si esta probabilidad no puede ser estimada, se listarán las posibles consecuencias, para que éstas puedan ser usadas en la evaluación de los riesgos y en el control de los riesgos.

El sistema usado para la categorización cualitativa o cuantitativa de la probabilidad de ocurrencia y de la severidad de un peligro debe estar registrado en el PGR.

2.3.2. Evaluación de los riesgos

Una vez identificados los riesgos conocidos o previsibles, el siguiente paso consiste en realizar una Evaluación de los Riesgos.

Para cada peligro identificado, el fabricante debe decidir, usando los criterios definidos en el PGR, si el peligro requiere una reducción del riesgo o no.

2.3.3. Control de los riesgos

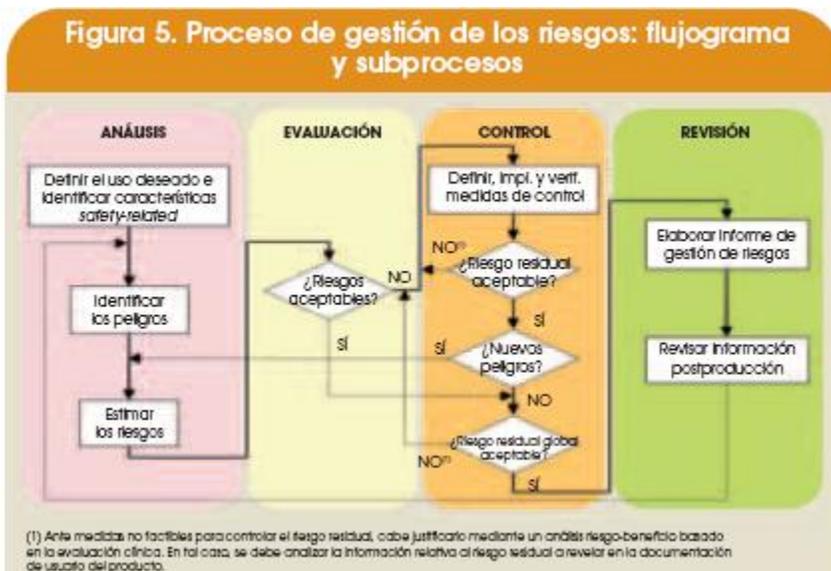
Una vez identificados los riesgos a mitigar, el siguiente paso consiste en identificar las medidas de Control de los Riesgos que sean apropiadas para reducir estos riesgos a un nivel aceptable.

Estas medidas de control deben responder a una o más de las siguientes opciones de control listadas en orden de prioridad:

- Seguridad inherente por diseño.
- Medidas protectoras en el propio producto sanitario o en el proceso de fabricación.
- Información para un uso seguro.

Una vez identificadas las medidas de control, el fabricante debe implementarlas y, tras ello, verificar su implementación, cerciorándose de su efectividad individual y de conjunto.

Cualquier riesgo residual, i.e. cualquier riesgo que prevalezca una vez tomadas estas medidas de control debe ser evaluado siguiendo los criterios definidos en el PGR. Si el riesgo residual se evalúa como no aceptable, será necesario establecer medidas de control adicionales y repetir el ciclo de reducción del riesgo.



Si el riesgo residual se evalúa como no aceptable y las medidas de control adicionales se muestran no factibles, el fabricante puede basarse en un Análisis Riesgo/Beneficio, es decir, en la recopilación de datos y literatura para determinar si los beneficios médicos del uso del producto superan con creces el riesgo residual como para permitir asumirlo.

En caso afirmativo, el riesgo residual pasaría a ser considerado aceptable. En caso negativo, el riesgo residual permanecerá inaceptable.

Se debe revisar en cada ciclo que los efectos de las medidas de control no generen nuevos peligros ni afecten a las estimaciones previas de otros riesgos. En caso afirmativo, será necesario repetir el ciclo de reducción de los riesgos.

Al final del proceso, el fabricante debe asegurarse de que todos los riesgos a controlar hayan sido considerados, y de que el riesgo residual global que presenta el producto sanitario es aceptable siguiendo los criterios definidos en el PGR o haciendo uso del Análisis Riesgo/Beneficio.

Finalmente, si este riesgo residual global se evalúa como aceptable, el fabricante debe decidir si procede revelarlo al usuario y, en caso afirmativo, la información necesaria a incluir en la documentación de usuario del producto.

2.3.4. Revisión e informe de los riesgos

El último paso, antes de liberar el producto para su comercialización, consiste en revisar el proceso de gestión de los riesgos, seguido para esta instancia o versión del producto que se va a liberar para asegurarse de que no se ha descuidado ningún riesgo o aspecto a considerar y emitiendo al efecto un Informe de Gestión de los Riesgos (ver sección 2.2.3).

2.3.5. Información de postproducción

El proceso de gestión de los riesgos no termina una vez liberado el producto sanitario; debe existir un paso adicional para recoger y revisar la información sobre el producto durante la fase de postproducción y realimentar así el proceso de gestión de los riesgos (ver sección 1.4).

3. Proceso de ingeniería de la usabilidad

3.1. Introducción

La norma IEC 62366 define un proceso para analizar, especificar, diseñar, verificar y validar la usabilidad, puesto que ésta contribuye a un uso seguro del producto sanitario. Este proceso de ingeniería de la usabilidad permite evaluar y controlar los riesgos causados por problemas de usabilidad asociados con el uso previsto o normal del producto.

Esta norma puede verse como una extensión de la norma ISO 14971, con la que converge en los puntos mostrados en la figura 6.



3.2. Expediente de ingeniería de la usabilidad

El proceso de ingeniería de la usabilidad debe materializarse en un Usability Engineering File

o Expediente de Ingeniería de la Usabilidad (ver figura 7), el cual debe incluir lo siguiente:

- La definición del proceso de ingeniería de la usabilidad.
- La parte del Expediente de Gestión de los Riesgos relativa a riesgos de usabilidad.
- La identificación de las Funciones Operativas Primarias (FOP).
- La especificación de la usabilidad.
- El Plan de Validación de la Usabilidad.
- El diseño de la interfaz de usuario (y lógicamente su implementación).
- La verificación de la usabilidad: sus resultados.
- La validación de la usabilidad: sus resultados.
- Los manuales de usuario.
- El training y el material del training.

3.3. Especificación de la usabilidad: funciones operativas primarias (FOP)

La norma remarca que el fabricante debe considerar las Funciones Operativas Primarias (FOP) del producto sanitario en su proceso de ingeniería de la usabilidad.

Las FOP son aquellas que conllevan una interacción del usuario y que, o bien están relacionadas con la seguridad en el uso del producto sanitario, o bien son frecuentemente utilizadas. Se incluyen estas últimas debido a que una usabilidad inadecuada de las mismas puede añadir sobrecarga al usuario y ser una fuente de frustración, incrementando la probabilidad de cometer errores de uso que afecten a la seguridad.

Las FOP servirán de entrada al proceso de Especificación de la Usabilidad para:

- Describir los escenarios de uso asociados a estas FOP.
- Especificar requisitos de interfaz de usuario para estas FOP, incluyendo aquéllos para mitigar riesgos.
- Especificar requisitos para determinar cuando las FOP son fácilmente reconocibles por el usuario.

3.4. Interfaz de usuario

El fabricante debe diseñar e implementar la interfaz de usuario del producto sanitario, dando así respuesta a la especificación de la usabilidad realizada. Para ello utilizará los métodos y técnicas de ingeniería de la usabilidad que le sean apropiados. Ejemplos de técnicas típicas de modelado de la interfaz de usuario, son las siguientes:

- diagrama de globos
- storyboard
- modelos: de bloque, de apariencia, de trabajo
- prototipos: de la interfaz del usuario, del producto sanitario.

Figura 7. Expediente de ingeniería de la usabilidad



3.5. Verificación y validación de la usabilidad

El fabricante debe verificar el diseño y la implementación de la interfaz de usuario, así como la usabilidad del producto sanitario en su conjunto, con respecto a la especificación de la usabilidad realizada.

Asimismo, debe realizar su validación conforme al Plan de Validación de la Usabilidad diseñado, el cual, típicamente, por tratarse de un producto sanitario, puede involucrar a usuarios externos y a personal especializado.

3.6. Manuales de usuario

Si se facilitan manuales de usuario junto con el producto sanitario, estos manuales:

- Deben incluir un resumen de la especificación de uso del producto.
- Deben incluir una breve descripción del producto, abarcando, si ello es relevante para el uso, lo siguiente: principio de funcionamiento, características físicas, características de funcionamiento y perfil del usuario previsto.
- Deben estar redactados en consonancia con el perfil del usuario previsto.
- Pueden facilitarse de forma electrónica. En este caso, el proceso de ingeniería de la usabilidad debe incluir una observación sobre qué información tiene que facilitarse también mediante copia en papel o marcada sobre el propio producto.

3.7. 'Training' y material del 'training'

Si se conoce o se intuye la necesidad de training a los usuarios acerca del producto sanitario, para asegurar un uso seguro y eficaz de al menos una FOP, el fabricante debe proporcionar este training, y/o el material del training a los usuarios. Al igual que los manuales de usuario, el training y el material del training deben ser acordes al uso previsto del producto y a los perfiles de los usuarios.

4. Procesos del ciclo de vida del 'software'

4.1. Introducción

La norma IEC 62304 especifica los requisitos para los procesos del ciclo de vida de sistemas software que sean en sí mismos productos sanitarios o que formen parte integrante de otros sistemas o dispositivos que sean, como conjunto, considerados productos sanitarios.

Es una norma armonizada por la UE y los EE. UU., por lo cual representa la referencia adecuada en lo que respecta a software para aplicación médica, para cumplir con los requisitos regulatorios de ambos mercados.

4.2. Clasificación de seguridad del 'software'

El fabricante debe asignar al sistema software una clasificación de seguridad. Esta clasificación está basada en los efectos que puede causar el producto sanitario sobre las personas que lo usen (pacientes, operadores, instaladores, mantenedores, personal sanitario, etc.), a los que el sistema software puede contribuir.

La clasificación de seguridad es la siguiente:

- Clase A: no es posible lesión o daño para la salud.
- Clase B: es posible una lesión no seria.
- Clase C: es posible una lesión seria o muerte.

Esta clasificación del sistema software, que debe ser coherente con la clasificación del

producto sanitario a efectos regulatorios (ver sección 1.3), se suele hacer constar también en el Expediente de Gestión de los Riesgos.

4.3. Requisitos por procesos del ciclo de vida del ‘software’

La clasificación de seguridad asignada al sistema sirve para graduar los requisitos a cumplir de la norma IEC 62304 para los procesos del ciclo de vida del software. Para la Clase C se exigen todos los requisitos de la norma, mientras que se exime de algunos a la Clase B, y de algunos más a la Clase A.

5. Conclusión

El fabricante de productos sanitarios, para poder comercializar sus productos, ha de dar respuesta a las regulaciones sanitarias relevantes en los mercados a los que se enfoca. Ello conlleva, aparte de seguir unos exigentes procesos de desarrollo y de producción, adquirir ciertos compromisos de interacción con las autoridades sanitarias, como los derivados del proceso de autorización para la puesta en el mercado (o para su renovación), y de los procesos de seguimiento y de vigilancia durante toda la fase de postproducción del producto.

El fabricante debe además permanecer alerta ante nuevos requisitos o posibles cambios en las regulaciones sanitarias de sus mercados, los cuales puedan implicar cambios sobre los procesos de desarrollo y de producción o sobre el propio producto, su documentación o marcado.

Mantenerse conforme a los estándares armonizados aquí presentados es la mejor garantía para permanecer alineado con las diferentes regulaciones y estar en la mejor disposición para afrontar los cambios que, sobre éstas, se produzcan.

Con todo, el esfuerzo a emplear por el fabricante puede no terminarse aquí. En efecto, la naturaleza, las características inherentes y el ámbito de aplicación de los productos sanitarios pueden exigir profundizar aún más hasta alcanzar la conformidad frente a estándares de aplicación muy técnica y/o específica.

Nuevos retos

Si algo caracteriza a la calidad del software y a sus profesionales es precisamente el deseo y el esfuerzo por la mejora continua. Pocos sectores como el de las TIC cambian y evolucionan de una manera tan rápida; y ello conlleva ponerse al día, seguir adelante con lo que se está haciendo y muchas veces iluminar al resto de la organización que se queda anclada a veces en su zona de confort.

Hay grandes expectativas y planteamientos para el nuevo milenio y no podía faltar alguien de nuestro comité hablando sobre ellos, tenemos los siguientes artículos:

- [Optimización energética en el ámbito de las TIC, Nueva cita en la agenda corporativa](#)
- [Una aproximación a los costes para conseguir la presencia en redes sociales](#)
- [Las redes sociales profesionales y el desarrollo de estándares en la industria del 'software' y la accesibilidad de Internet](#)
- [La revolución social en las empresas](#)



personas



procesos



tecnologías

Optimización energética en el ámbito de las TIC, Nueva cita en la agenda corporativa

RSE - GESTIÓN SOSTENIBLE / ARTÍCULO

Optimización energética en el ámbito de las TIC

Nueva cita en la agenda corporativa

Autor: Eva Carro Solana
Consultora de IT Governance de Atos Consulting. Empresa miembro del Comité de Software de la AEC

Publicado en la Revista CALIDAD - N° I • 2009



Texto Eva Carro Solana
Consultora de IT Governance de Atos Consulting. Empresa miembro del Comité de Software de la AEC



Introducción

Aunque no existe una definición de Green IT hasta la fecha, se puede definir como "una aproximación holística a la cultura medioambiental, la gobernanza sostenible y la gestión de la organización (alineación de TI y negocio), además de sus procesos y proyectos" (Cornè de Graaf, 2008).

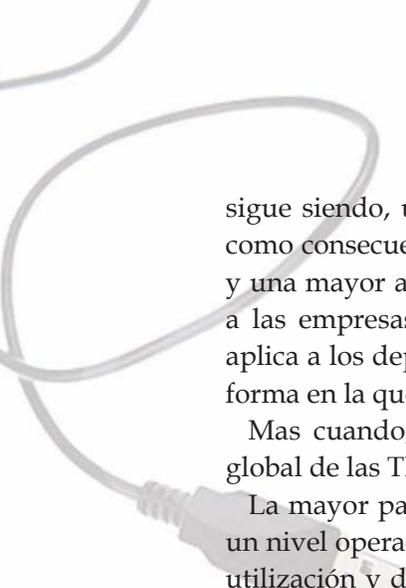
Se considera Green IT (también conocido como Green Computing, o informática verde) al estudio y práctica del uso eficiente de los recursos informáticos, en términos de capacidad y energía. Típicamente, los sistemas tecnológicos o los sistemas informáticos que consideran criterios ecológicos tienen en cuenta: la viabilidad económica, la responsabilidad social y el impacto ambiental. Esto difiere de las soluciones tradicionales de negocio que sólo tenían en cuenta la viabilidad económica de una solución.

Atendiendo a la práctica, las empresas han tenido dos tipos de acercamiento a la gestión de Green IT: uno a nivel más operacional, el mayoritario, y otro, un acercamiento a un nivel más holístico o estratégico, que ha sido el menos abordado. El primer nivel llamado operacional atiende más a la introducción de una serie de iniciativas específicas, generalmente en el departamento de Tecnologías de la Información (TI), mientras que el nivel estratégico atiende a la elección de un camino común para toda la organización. Se puede comparar el nivel operacional con sus iniciativas concretas a los objetivos específicos del modelo de madurez CMMI que abordan la implementación, o incluso comparar este nivel con el área operativa de modelos como ITIL, donde los procesos del día a día estaban más encaminados a "apagar fuegos", porque sin lugar a dudas la operación es más reactiva que proactiva. Por el contrario, el nivel estratégico es un nivel proactivo de Green IT que pretende integrarse en la estrategia de negocio.

A este nivel de gobernanza se necesita alinear la estrategia de Green IT con las estrategias e iniciativas del negocio.

Se entiende que adoptar Green IT no puede hacerse de una manera aislada sino que debe ser parte de la estrategia total de la empresa.

Como se ha comentado, y sin quitarle importancia, la práctica común de Green IT ha sido, y



sigue siendo, un conjunto de iniciativas dispersas que muchas organizaciones han adoptado como consecuencia de una mayor sensibilización hacia el medio ambiente, el cambio climático y una mayor atención a aspectos ecológicos. Ese aumento de conciencia medioambiental lleva a las empresas públicas y privadas a tomar responsabilidad en el asunto. Esto también se aplica a los departamentos de TI que debido a esa presión ecológica tienen que replantearse la forma en la que las TIC son gestionadas y gobernadas.

Mas cuando, según estimaciones de Gartner (firma de análisis de mercado), la industria global de las TIC es responsable del 2% de las emisiones globales de CO₂.

La mayor parte de las iniciativas para reducir las emisiones de CO₂ se han llevado a cabo a un nivel operacional, es lo que se ha llamado “ciclo de vida del equipo”, que abarca la compra, utilización y disposición del equipamiento informático. Y dentro de este nivel operacional, la mayor iniciativa asumida por las organizaciones ha sido el reciclaje tanto de papel, PC y otros dispositivos, como tóner.

A continuación se describe la gestión de Green IT a un nivel operacional, aplicado en centros de datos por su gran consumo energético. Posteriormente, se habla de Green IT a un nivel más profundo (holístico o estratégico), alineado con el negocio donde se resumen los pasos para adoptarlo en una organización.

Green IT desde un nivel operacional: con especial atención a los centros de datos

Los principios que utiliza Green IT a este nivel son similares a los de la “química verde”: reducción del uso de materiales peligrosos, como el plomo en las fases de manufactura, reciclaje, maximización de la eficiencia de la energía a lo largo del ciclo de vida de un producto, y maximización del grado de reciclado o biodegradación de los productos desechados tanto por usuarios como por fábricas.

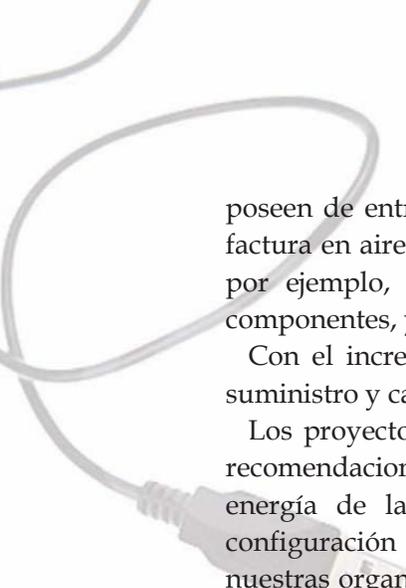
Green IT cubre tres aspectos fundamentales:

- La reducción del uso de materiales peligrosos en la fabricación de productos tecnológicos. Iniciativas al respecto: Cumplimiento y adaptación de ciertas normativas
- El reciclado o la biodegradación de los productos tecnológicos. Algunas iniciativas al respecto son: Cumplimiento y adaptación a ciertas, normativas, Recogida de máquinas obsoletas, Programas de donación y reutilización
- La optimización de la eficiencia energética en su uso. Área donde las empresas usuarias tienen muchas posibilidades de mejora en la eficiencia energética de los centros de datos.

En torno al 10-20% del consumo eléctrico hoy en día se destina a centros de datos con un crecimiento sumamente agresivo. Gartner señaló que la mayor organización de TI gasta el 5% de su presupuesto total de TI en energía, y esto podría elevarse dos o tres veces en los próximos cinco años.

El aire acondicionado es uno de los principales componentes del gasto energético de un centro de datos, habitualmente en torno al 50%. De ahí que cuestiones como la forma de los servidores sean fundamentales. Los blade servers, por ejemplo, permiten aprovechamientos energéticos muy superiores mediante la centralización de componentes como la refrigeración o la alimentación ininterrumpida (UPS): máquinas en un sólo rack, que se enfría, además, de una manera más eficiente.

El diseño de los flujos de aire en el centro de datos o el mantenimiento de condiciones de humedad adecuadas son factores muy importantes, condicionados por la localización geográfica. Países como Irlanda, con condiciones de humedad y temperatura adecuadas,



poseen de entrada mejores condiciones que un lugar en medio de un desierto, y reducen la factura en aire acondicionado o humidificación. Otros factores de interés son el uso de Linux, por ejemplo, que permite un control de las necesidades energéticas del kernel y otros componentes, y mejora las posibilidades de ahorro.

Con el incremento de los costes de energía y con los problemas crecientes de fiabilidad, suministro y capacidad, la electricidad necesita una estrategia específica propia.

Los proyectos de optimización de rendimiento y reducción de costes son una parte de las recomendaciones diarias en casi todas las áreas de negocios, y ahora se trata a los costes de energía de la misma manera. Los profesionales de TI toman decisiones acerca de la configuración e instalación de servidores, las especificaciones de los equipos que compran nuestras organizaciones y los requisitos para mejorar y crear centros de datos, incluso ofrecen información para el diseño inicial durante el desarrollo de aplicaciones. En lo que se refiere a estos proyectos, tienen una excelente oportunidad de ser ecológicos e influir en la eficiencia de la energía de cualquier centro de datos.

La primera parte de cualquier estrategia consiste en conocer cuál es el mapa de utilización actual de la energía. Es necesario saber dónde se usa la energía y en qué equipos específicos, así como qué uso es eficiente y cuál se desperdicia en el centro de datos. Desgraciadamente, es poco habitual encontrar un medidor de consumo de energía que pueda desglosar el uso a un nivel en que se puedan ver los resultados de sus acciones. Normalmente, la mayoría de empresas sólo ven una factura de energía mensual que acumula el consumo en el total de la línea, y esto supone un estímulo muy pequeño para el ahorro de energía, porque hace difícil apreciar la repercusión de las decisiones que se toman, y por tanto demostrar que los pequeños cambios introducidos revierten en ahorros de energía.

A continuación se presentan varias actuaciones para la optimización de la eficiencia energética.

Consumo eficaz de energía

Cambios recomendables:

- Analizar hasta qué punto un centro de datos sigue las recomendaciones que se presentan en un estudio, como el realizado por los laboratorios Lawrence Berkeley para el Consejo de EE UU para la promoción de una economía basada en la eficiencia de la energía (éste detalla acciones para mejorar la eficiencia de la energía desde la perspectiva de las instalaciones).
- Ajustar el número de servidores, memoria, discos e interfaces de red.
- Ajustar la capacidad del hardware.

El consumo de energía es un indicador que la dirección puede usar para medir la eficacia de decisiones tomadas respecto al ciclo de vida del hardware de TI. Se tiene que atender al consumo de energía en la infraestructura global de la empresa e involucrar a proveedores, productos, servicios, operaciones de la empresa, etc.

Supervisión de energía

Cambios recomendables:

- Desglosar el consumo de energía en distintas unidades de negocio, creando la percepción de que los costes de energía los controla la unidad de negocio.
- Instalar un medidor de energía y un sistema de monitorización e informes requiere una inversión inicial considerable. Sin embargo, invertir en este tipo de sistemas conlleva

como resultado un ahorro de costes mayor.

- Es necesario crear un plan para determinar cómo se usarán los datos recopilados para una mejor administración del consumo de energía y obtener, al final, un retorno de inversión medible.

¿Cuánta energía necesita ahorrar para justificar una actualización del sistema de supervisión de energía en su centro de datos? Ser conscientes que, debido al enorme gasto de estos centros, las mínimas mejoras de eficiencia en ellos pueden representar ahorros de costo sustanciales.

Analizar dónde está el ahorro

Se deben considerar muchos factores para determinar el ahorro. ¿Cuál es el coste de agregar más energía en un área determinada? ¿Existe una estrategia para minimizar las restricciones de energía? ¿Es posible resolver el problema ubicando los centros de datos en lugares donde la energía sea más económica? ¿Es mejor alternativa centrarse en operaciones que mejoran las instalaciones existentes?

Hay acciones menos “drásticas”. La mayoría de los departamentos de TI no saben cuánta energía consumen sus sistemas en la actualidad, lo que comporta una buena oportunidad para un posible ahorro. Un primer paso podría ser examinar el consumo de energía asociado a dispositivos individuales y, a continuación, ver si existen cambios que se puedan realizar por dispositivo. Muchos cambios pequeños pueden sumar ahorros sustanciales.

Es posible agregar medidores a los bastidores del centro de datos y recopilar los de distintos dispositivos. Las herramientas de administración se pueden usar para recopilar datos directamente desde algunos servidores con capturas vía SNMP (Simple Network Management Protocol) del consumo de energía.

Virtualización

Con la virtualización, un administrador de sistemas podía combinar diferentes sistemas físicos y máquinas virtuales en un único y potente sistema “desconectando” el hardware original y, por tanto, reduciendo el consumo de energía y facilitando la refrigeración. Muchas compañías comerciales y proyectos de código abierto ofrecen ahora paquetes de software que permiten la transición virtual. La virtualización puede ahorrar hasta un 50% de los gastos operativos concurrenciosos.

Gestión de la energía

Un estándar abierto llamado ACPI (Advanced Configuration and Power Interface) provee de un interfaz de programación estándar que permite a un sistema operativo controlar directamente los aspectos de consumo y ahorro de energía del hardware sobre el que se ejecuta.

Esto permite al sistema desconectar automáticamente componentes como monitores y discos duros después de determinados periodos de inactividad.

Además, un sistema puede hibernar, proceso en el que desconecta prácticamente todos sus componentes, incluyendo la CPU y la RAM del sistema, reduciendo significativamente el consumo de energía del sistema.

Per se, los principales sistemas operativos cuentan con herramientas y aplicaciones de control y gestión de energía que ayudan a controlar el consumo de energía de los ordenadores.

Nuevo hardware



Desde 2007, al menos cinco fabricantes de ordenadores (Everex, Linutop, Systemax, Zonbu y DELL) comercializan PC de bajo consumo. Dell, siguiendo la guía del programa Energy Star de la EPA, ha creado varios ordenadores de sobremesa y portátiles que consumen menos de cinco vatios en modo de bajo consumo.

Otra manera de limitar el consumo de energía es eliminar el uso de salvapantallas, usar monitores LCD, usar sistemas de alimentación certificados 4.0 (eficiencia del 80% como mínimo), switches y routers que reduzcan los costes operativos, entre otras.

Reciclaje de materiales

Equipos de informática obsoletos pero que aún funcionan pueden ser donados a diversas organizaciones sin ánimo de lucro. Los sistemas rotos o tan antiguos que no son aceptados por estas organizaciones pueden reciclarse en algunos centros comerciales, tiendas de informática, de electrónica y en centros gubernamentales o privados de reciclaje.

El reciclaje de sistemas informáticos evita que materiales nocivos como el plomo, mercurio o cromo hexavalente sean desechados. De todos modos, muchas veces los sistemas recogidos en programas de reciclaje son enviados a países en vías de desarrollo, donde los estándares ambientales son menos estrictos que en Europa y Norteamérica. La Silicon Valley Toxics Coalition estima que el 80% de la e-basura reciclada en Estados Unidos, Europa y Japón es enviada a países como India, China o Pakistán.

Fuentes de energía alternativa (verde)

Los ordenadores pueden alimentarse de diferentes fuentes de energía procedentes de diversos sistemas de generación: eólica, hidroeléctrica, paneles fotovoltaicos, nuclear, etc.

Green IT desde un nivel estratégico

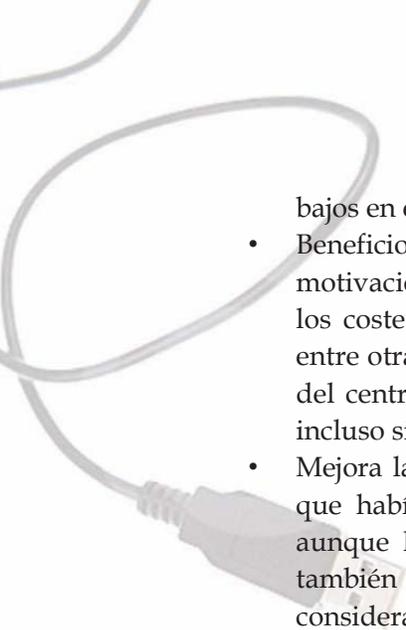
Green IT no sólo necesita ser adoptada sino que también necesita ser gestionada y gobernada correctamente dentro de la estrategia de TI. Introducir medidas técnicas es una parte de la solución.

Como elemento integrante de toda estrategia de gobernanza de TI, la Green IT cubre varios aspectos del gobierno corporativo, como políticas de energías o políticas de RR HH referentes al trabajo en casa. Incluye iniciativas para acortar el tiempo que se tarda en llegar a la oficina o reducir los viajes de trabajo mediante la estimulación del trabajo en casa, videoconferencias, etc. Éstas no son iniciativas propias de TI pero sin duda impactan en ella. Desde esta perspectiva, Green IT pretende superar la alineación de los objetivos “verdes” de TI para integrarlos en los objetivos de negocio. Y, al igual que lo harían los objetivos genéricos en un modelo como CMMI, Green IT busca la institucionalización de sus iniciativas.

En este nivel de gestión se realizaría un boceto de la estrategia “verde” en consonancia con la estrategia de negocio, se designaría algún responsable para Green IT, se activaría la gestión del coste de energía, incluso se establecerían acuerdos “verdes” y SLA con KPIs “verdes”, etc.

Los beneficios de introducir Green IT

- Reducir la contaminación, la emisión de CO₂ y el nivel de e-basura. Investigaciones de la consultora medioambiental británica Carbon Trust y de la prestigiosa institución académica London Imperial College demuestran que para el 2050 se podrían reducir en un 60% las emisiones de carbono, utilizando una combinación de eficiencia energética, recursos eléctricos renovables, sustituyendo el carbón y el petróleo por combustibles más



bajos en emisiones de CO₂, como el gas y con el uso de hidrógeno como combustible.

- Beneficios en términos de ahorro de costes. Según IDC's Green Survey, la primera motivación o impulsor de la adopción de Green IT se debe a motivos económicos, reducir los costes operativos como: la virtualización, el trabajo en casa, las videoconferencias, entre otras muchas acciones, implican un considerable ahorro de costes. La optimización del centro de datos lo hará más eficiente y más fácil de gestionar; así Green IT puede incluso significar un diferenciador competitivo.
- Mejora la imagen de la organización. En un estudio realizado en Bélgica con empresas que habían tomado medidas para reducir sus emisiones de CO₂ se descubrió que aunque la mayoría informa de esa iniciativa internamente, sobre el 40% indica que también lo hace externamente mediante prensa, informes anuales, etc. Esto es considerado como una muestra de que las empresas en general están deseando compartir información sobre sus actividades "verdes" a todo el mundo.
- Aumentar valor para accionistas. Lo expuesto en el párrafo anterior contribuye a la imagen "verde" de la empresa y a crear seguridad y valor para los accionistas.
- Contribuir positivamente a la total gobernanza de TI. Al añadir valor a la imagen de la compañía como un todo, Green IT contribuye de forma positiva a la gobernanza total de TI y a la gobernanza de la organización.
- Representa una ventaja competitiva. Como proveedor será considerado green partner, como empresa estará satisfaciendo a los usuarios (clientes) que demandan productos y servicios "verdes", porque el departamento de TI habrá cumplido con las expectativas de éstos. Según IDC's Green Survey, más del 50% de los clientes tienen en cuenta las empresas "verdes" cuando eligen un proveedor TIC. Un tercio de los clientes ya considera importante o muy importante que los proveedores TI tengan ofertas "verdes". Green IT formará parte de muchos proyectos TI en los que se trabajará junto a terceras partes (como proveedores de hardware/software, subcontratistas). Y en el futuro, el criterio típico de adquisición o compra incluirá criterios de Green IT, integrados en requisitos "verdes" de ofertas (RFP).

La gestión de los proveedores "verdes" (o gestión de terceros "verdes") se convertirá en un tópico. Es probable que incluso se llegue a definir indicadores de niveles de servicio "verdes" como parte de los SLA acordados con terceros.

- Adelantarse a normativas ambientales más estrictas.

La tendencia es que se endurezcan las leyes medioambientales, con lo que adoptar cuanto antes Green IT es una buena opción.

La gestión del riesgo de Green IT

Como en cualquier proyecto la ejecución de Green IT se balancea entre el riesgo y el retorno de la inversión. Se tiene que considerar el riesgo por cada iniciativa de Green IT y el beneficio potencial de la ejecución. Otro elemento de riesgo es la legislación, con la tendencia existente el riesgo del no cumplimiento de normativas es muy alto; lo recomendable es gestionarlo de forma proactiva.

Mediciones

Como en cualquier implantación de un programa se necesitan métricas que reflejen el progreso en su ejecución y el cumplimiento de los objetivos. Ser capaz de medir el estado actual y futuro de Green IT permitirá a la organización monitorizar el progreso y tomar acciones correctivas si fuese necesario.

La medición de Green IT incluye tanto medidas cuantitativas como cualitativas.

Las medidas cuantitativas son principalmente derivadas de datos operativos como el uso de energía de un centro de datos. Sin embargo, son igual de importantes las medidas cualitativas porque complementan a las cuantitativas. Ejemplo de ello son el nivel de concienciación, la adopción de Green IT en una estrategia formal, el nombramiento de un responsable Green IT (Green IT role), etc.

Plan de actuación (pasos para la adopción de Green IT)

1. Conseguir el compromiso de la alta dirección. Si ésta no se convence de la necesidad de Green IT, los cambios serán limitados. Green IT es una iniciativa estratégica y necesita un patrocinio para empezar (éste puede ser el CIO pues según Gartner Green IT ha sido la prioridad número uno para el CIO en 2008).
2. Los departamentos TIC deben definir una política medioambiental y desarrollar una estrategia que acometa las acciones necesarias para reducir el impacto medioambiental de las infraestructuras TIC y las operaciones diarias de la empresa (efectos de primer orden).
3. La estrategia debería también incluir acciones para reducir el impacto medioambiental de los productos (comercializados) y servicios (prestados) de la empresa, así como su cadena de suministros (efectos de segundo orden).
4. Atribuir el rol de propietario de Green IT a alguien que lidere y coordine las iniciativas de Green IT. Será responsable de su implementación y deberá reportar a la dirección quien es el responsable último de la implementación.
5. Conocer dónde estamos con una rápida evaluación o una evaluación más detallada similar a una auditoría.
6. Gestión del cambio cultural. Es necesario implicar a los empleados en un programa de concienciación medioambiental. La mayoría de empresas que han lanzado programas medioambientales han obtenido una respuesta positiva del personal. El proceso de gestión del cambio ha de comprender una serie de acciones como apagado de equipos informáticos, impresión responsable, etc.
7. Elaborar un plan de acción o road map basado en la evaluación. Éste recogerá los puntos de mejora donde intervenir signifique una mejora rápida y otras metas más ambiciosas. Incluirá formación y campañas de concienciación.
8. Realizar un control y seguimiento del plan de acción. Además, la adopción de Green IT no se logra de la noche a la mañana, es un proceso continuo que debemos abordar como el modelo de mejora continua de la gobernanza de las tecnologías de la información.

Notas de prensa:

Noviembre 2008. De acuerdo con los resultados de un reciente sondeo de Forrester Research, la mayoría de las empresas tienen intención de mantener su inversión en proyectos de Green IT a pesar del actual clima de incertidumbre económica. El principal objetivo perseguido por estos proyectos es la reducción de costes.

Septiembre 2008. Greenpeace publica la 9ª edición del ranking verde de empresas de productos electrónicos el cual informa sobre el comportamiento de las empresas líderes en la fabricación de ordenadores, teléfonos móviles, televisores y videoconsolas respecto a sus políticas y prácticas globales para la eliminación de productos químicos peligrosos y la responsabilidad que asumen sobre sus productos una vez que éstos han sido desechados por

los consumidores. Nokia aparece en cabeza con una puntuación de 7 sobre 10.

Octubre 2007. El alemán Peter Grünberg y el francés Albert Fert fueron honrados con el premio Nobel de Física 2007 por su descubrimiento de la magnetoresistencia gigante, que revolucionó la tecnología del almacenamiento de datos.

Octubre 2007. El 42% de las empresas desconoce el gasto energético que genera su infraestructura tecnológica.

Septiembre 2007. Se calcula que sólo el 10% de todos los ordenadores que se desechan se reciclan.

Junio 2007. Google e Intel fundan la Climate Savers Computing Initiative, una iniciativa con el objetivo de reducir el consumo eléctrico de ordenadores en estado activo y suspendido.

Febrero 2007. Se inaugura The Green Grid, un grupo integrado por diversos suministradores tecnológicos que colaborarán para mejorar la eficiencia energética de los centros de datos. El grupo anima a los usuarios empresariales a unirse a la iniciativa.

Documentación de referencia y fuentes:

Advanced Micro Devices (www.amd.com/es-es/)

Business Green (www.businessgreen.com/)

Carbon Trust (www.carbontrust.co.uk/)

Climate Savers Smart Computing (www.climatesaverscomputing.org/)

Computer World (www.computerworld.com)

Desarrollo Inteligente (www.desarrollointeligente.org/)

Deutsche Welle (www.dw-world.de/dw)

Energy Star (www.energystar.gov/)

Gartner (www.gartner.com)

Green IT Blog (www.greenitblog.com/)

Greener Computing (www.greenercomputing.com/)

Greenpeace (www.greenpeace.org/)

HP (welcome.hp.com)

IBM (www.ibm.com/us/en/)

IDC (www.idc.com/)

Intel (www.intel.com)

Internacional Data Group (www.idg.com)

International Energy Agency (www.iea.org/index.asp)

Linux (www.linux.org/)

TCO (www.tco.se/)

Una aproximación a los costes para conseguir la presencia en redes sociales

ARTÍCULO / REDES SOCIALES



Una aproximación a los costes para conseguir la presencia en redes sociales



Juan Antonio Caloto
Director de TI de la AEC
Secretario del Comité CSTIC
de la AEC

Por fin nos hemos decidido a tener presencia en las redes sociales. Por un motivo u otro, nos hemos convencido de la oportunidad que nos dan, de conocer lo que se habla de nosotros, de potenciar las relaciones con nuestros clientes, de fidelizar a nuestra marca, de tener más visibilidad y, en definitiva, de tener la posibilidad de generar más negocio. Esta decisión también puede venir motivada por el hecho de que nuestros clientes están allí y no estar nosotros puede salirnos caro.

Tomar la decisión es habitual hoy en día en las organizaciones, bien por probar o por estar totalmente convencidos. Pero eso no basta: tienen que transformar esa decisión en una realidad a través del establecimiento de unos objetivos, una estrategia y una planificación de actividades que impliquen unos costes.

En definitiva, se tendrá que abordar como cualquier otro proyecto en la

organización. Tendrá que definir unos objetivos: ¿qué fines persigue?, ¿dónde quiere llegar?... y la estrategia para conseguirlos: ¿dónde decide estar? ¿De qué forma? ¿Con qué medios?...

Como cualquier otro proyecto tendrá unos costes de desarrollo y se realiza para obtener unos beneficios, aunque éstos puedan no ser directamente económicos. Además, deberá o bien tener conocimientos suficientes dentro de ella o bien contratar ayuda externa o formarse, dedicando tiempo al aprendizaje (tipo de redes, funcionamiento, formas de interacción); posteriormente, después de un análisis de la situación de partida, deberá planificar, desarrollar la actividades necesarias, analizar y mejorar.

Sin embargo, no debemos caer en el error de pensar que esto es sencillo y gratis. Sí es cierto que muchas de las redes sociales son de acceso libre y gratuito y también es cierto que hay un amplio surtido de herramientas de social media

Autor: Juan Antonio Caloto Director de TI de la AEC, Secretario del Comité CSTIC de la AEC

Publicado en la Revista CALIDAD - Nº IV • 2012

Por fin nos hemos decidido a tener presencia en las redes sociales. Por un motivo u otro, nos hemos convencido de la oportunidad que nos dan, de conocer lo que se habla de nosotros, de potenciar las relaciones con nuestros clientes, de fidelizarlos a nuestra marca, de tener más visibilidad y, en definitiva, de tener la posibilidad de generar más negocio.

Esta decisión también puede venir motivada por el hecho de que nuestros clientes están allí y no estar nosotros puede salirnos caro.

Tomar la decisión es habitual hoy en día en las organizaciones, bien por probar o por estar

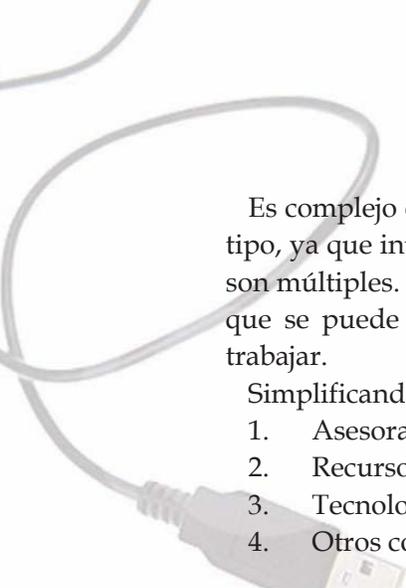
totalmente convencidos.

Pero eso no basta: tienen que transformar esa decisión en una realidad a través del establecimiento de unos objetivos, una estrategia y una planificación de actividades que impliquen unos costes.

En definitiva, se tendrá que abordar como cualquier otro proyecto en la organización. Tendrá que definir unos objetivos: ¿qué fines persigue?, ¿dónde quiere llegar?... y la estrategia para conseguirlos: ¿dónde decide estar? ¿De qué forma? ¿Con qué medios?...

Como cualquier otro proyecto tendrá unos costes de desarrollo y se realiza para obtener unos beneficios, aunque éstos puedan no ser directamente económicos. Además, deberá o bien tener conocimientos suficientes dentro de ella o bien contratar ayuda externa o formarse, dedicando tiempo al aprendizaje (tipo de redes, funcionamiento, formas de interacción); posteriormente, después de un análisis de la situación de partida, deberá planificar, desarrollar la actividades necesarias, analizar y mejorar.

Sin embargo, no debemos caer en el error de pensar que esto es sencillo y gratis. Sí es cierto que muchas de las redes sociales son de acceso libre y gratuito y también es cierto que hay un amplio surtido de herramientas de social media que son "gratuitas", pero habitualmente sólo lo son hasta un cierto punto. Puede darse el caso de que por el tamaño, la complejidad de la organización o las necesidades iniciales que tenga, no necesite más. Pero también puede requerir funciones más avanzadas u otro tipo de herramientas más complejas que sean de pago. Todo ello sin dejar de tener en cuenta el tiempo de dedicación del personal en el proyecto.



Es complejo definir el coste que podrá tener una organización al abordar un proyecto de este tipo, ya que intervienen muchos factores y las decisiones que pueden tomar las organizaciones son múltiples. Pero sí se puede hacer un planteamiento de las principales áreas de costes en las que se puede incurrir, para así dar una idea de partida sobre la que se puede empezar a trabajar.

Simplificando, podremos plantear las siguientes áreas de coste:

1. Asesoramiento, formación y aprendizaje.
2. Recursos humanos.
3. Tecnología: hardware y software.
4. Otros costes.

Asesoramiento, formación y aprendizaje

La organización debe tomar la decisión de si aborda el proyecto exclusivamente con recursos internos o requiere apoyo externo. En esta decisión pesan numerosos factores: tamaño de nuestra organización, plazos planteados, disponibilidad de conocimiento interno, de tiempo, de recursos, de dinero... Al tomar esta decisión, la organización puede decantarse por ir a un extremo u otro o bien quedarse en alguna de las posibles situaciones intermedias que se puedan dar. En general, lo que suele suceder es alguna situación intermedia: desarrollo del proyecto con recursos internos y apoyo externo. El peso que tenga cada uno dependerá de la decisión que tome la organización. En cualquier caso, en función de esta decisión, para evaluar el coste, la organización deberá considerar:

- Las necesidades que tenga de formación, tanto para el personal directamente implicado como para el resto de la plantilla, que considere oportuno. Tendremos que analizar la formación requerida para cada perfil y puesto de trabajo. Aquí podrán entrar desde actividades de formación presencial a formación online, programa de sensibilización interna, hasta modificaciones que se realicen en el manual de acogida de la organización.
- Cuando se recurre a colaboraciones externas, éstas no sólo vendrán como apoyo en temas de social media, sino que se integrarán en la mayoría de las situaciones dentro de campañas globales de marketing. Por otro lado, hay que contar con el posible asesoramiento legal externo que pueda necesitar sobre la actividad que va a desarrollar en este nuevo entorno.

En el caso de una organización pequeña, por ejemplo, lo deseable puede ser contar con algún experto que le ayude a arrancar y luego, poco a poco, lograr cierta independencia y contar entonces únicamente con ayudas puntuales.

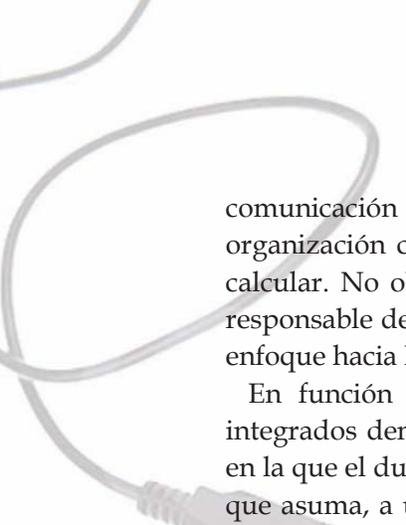
Recursos humanos

Como ya hemos comentado la organización necesitará personal dedicado a tiempo completo o parcial al proyecto.

De esta forma, tendrá que desarrollar diversos roles:

- Community Manager o Responsable de Comunidad o gestor de comunidad.
- Social Media Analyst o Analista de Social Media.
- Social Media Strategist o Estratega de Social Media.

No es necesario tener definidos estos roles en la organización. Se puede plantear otra estructura o clasificación, según sus necesidades, pero sí deberá tener en cuenta cada uno de los tres aspectos que representan los roles anteriormente mencionados: la comunicación y el contacto, el análisis y la estrategia. Puede darse la situación, por ejemplo, de que la parte de



comunicación y contacto esté repartida entre toda o una gran parte del personal de la organización con actividad en redes sociales. Los costes en este caso son más complejos de calcular. No obstante, se podrá realizar una estimación. Eso sí, siempre deberá haber algún responsable de proceso, social media, que tenga la visión general de la situación y transmita el enfoque hacia los objetivos que se persiguen, así como perfiles de apoyo.

En función del tamaño y complejidad de la organización, estos perfiles podrán estar integrados dentro de una sola persona o de varias. Podemos ir desde una micro-micropyme, en la que el dueño es el empleado y lo realiza todo, por lo que ésta será una nueva función más que asuma, a una gran empresa, en la que participará personal de diversos departamentos y podrá haber diversos perfiles por línea de negocio, tipo de servicio...

Al calcular los costes, la organización debe tener claro que es una actividad que tendrá una dedicación de tiempo todos los meses y, en función de las campañas o acciones que se desarrollen, podrá variar o ser constante.

Tecnología

'Hardware'

En este proyecto, la mayor parte de la actividad que desarrolle la organización será en los entornos virtuales que proporcionan las redes sociales, por lo que será imprescindible contar con equipos de trabajo, desde un PC a múltiples ordenadores con dedicación parcial o completa, en función de la actividad.

La organización deberá valorar la importancia que tienen en este caso, la movilidad y la disponibilidad. Las redes sociales no cierran cuando se acaba el horario laboral, ni se puede dejar un mensaje diciendo que se responderán los comentarios cuando vuelva del viaje la persona responsable, tal y como se utiliza en ocasiones en las comunicaciones por correo electrónico.

A la vez, las redes sociales demandan información en vivo, y cualquier organización se puede encontrar con situaciones en las que necesita transmitir eventos en directo en otros lugares que no son donde están los equipos de trabajo habitual.

Por todo ello, se deberá valorar la necesidad de portátiles, tablet, smartphone... Ciertamente es que estos últimos suelen ser ya habituales en cualquier reunión o evento.

Todo esto requerirá unas necesidades de conectividad a red corporativa que, en el caso de pequeñas organizaciones, puede ser "muy sencillo", pero en organizaciones grandes se pueden dar situaciones complejas a nivel de seguridad.

'Software'

Enlazando con el apartado anterior, el software que use la organización deberá mantener las mismas características de accesibilidad y disponibilidad, por lo que el planteamiento suele ser que esté disponible en la "nube", especialmente la parte de la gestión de conversaciones.

Por lo general, como ya hemos comentado, existen buenas opciones gratuitas que permitirán gestionar redes sociales de una forma muy profesional,

pero en cuanto se avanza un poco en optimizar los tiempos de trabajo a la hora de, por ejemplo, publicar en varias redes, programación de noticias o cualquier otra optimización de este tipo, será necesario avanzar hacia opciones de pago que tienen las mismas herramientas que mencionábamos antes, o puede ser necesario avanzar hacia opciones más potentes. La opción más básica sería la utilización de los clientes nativos, que suelen proporcionar cada una

de las redes sociales.

Algunos ejemplos:

- Gestión de conversaciones, coordinación de equipos y programación de noticias: Hootsuite.
- Gestión de conversaciones y programación de noticias: Tweetdeck.
- Programación de noticias: Buffer.
- Monitorización: SocialBro, Social Mention, Pirendo...
- Automatización: IFTTT.
- Recopilación de ideas y recortes de investigación: Evernote, Springpad...
- Clipping de noticias: Google Reader, Paper.li, Scoop.it, Flipboard...
- Creación de una tienda en Facebook: Social-Buy o Codeeta....

Por otro lado, existen herramientas de alta gama para la gestión y monitorización, las cuales se pueden denominar "Social CRM", donde los líderes de mercado, según la consultora tecnológica Gartner, son Jive, Salesforce.com, Lithium.

La organización debe decidir qué herramientas pueden serle más útiles. No existe la herramienta perfecta y si se opta por una herramienta a medida, los costes tienden a incrementarse.

Otros costes

Dentro de esta partida, la organización puede englobar todo aquello que no ha añadido en los gastos anteriores:

- Diseño web y creatividades:
 - Diseños de blog corporativos.
 - Personalización de las diferentes páginas de redes en las que decida tener presencia: Fan Page en Facebook, página de Twitter, Google +, Foursquare...
 - Imágenes tanto para las personalizaciones como para las noticias y artículos que se desarrollen.
- Gasto en publicidad offline, todo tipo de material promocional de la actividad: cartas, revistas, regalos y promociones.
 - Infografías.
 - Vídeos.
 - Logos...
- Gasto en publicidad online, tanto los enfocados a la optimización en buscadores como en redes sociales, o bien los enfocados a marketing en ambos entornos (SEO/SMO⁷, SEM/SMM⁸).
- Elaboración de contenidos para blog, encuestas, documentos, redacción de noticias...
- Optimización de la web corporativa y correlación con las páginas creadas en las diferentes redes.
- Otros costes vinculados a la actividad (desplazamientos, dietas, alojamientos, fiestas, eventos...).

Una vez analizado todo esto, podríamos tener una estimación de los gastos iniciales, así como de los gastos de mantenimiento. Posteriormente, en caso de abordar el proyecto, la organización tendrá que analizar si ha cubierto los objetivos perseguidos, pero eso ya es otra historia.

⁷ SEO o Search Engine Optimization (optimización para los motores de búsqueda) / SMO o Social Media Optimization (optimización para medios sociales).

⁸ SEM o Search Engine Marketing (Marketing en motores de búsqueda) / SMM o Social Media Marketing (Marketing a través de los medios sociales).

Las redes sociales profesionales y el desarrollo de estándares en la industria del 'software' y la accesibilidad de Internet



Autor: Félix A. Barrio PhD, CISM Gerente de Innovación y Calidad del Instituto Nacional de Tecnologías de la Comunicación (INTECO) y Vocal del Comité de CSTIC de la AEC

Publicado en la Revista CALIDAD - Nº IV • 2012

La importancia de las redes sociales para la promoción de las normas o estándares de promoción de la mejora y el aseguramiento de la calidad se revela desde hace casi una década como trascendental. Esto es así porque las redes sociales facilitan una interacción mucho más intensa y permanente entre aquellas personas

interesadas en promover la normalización; en primer lugar, al servir de instrumento a la permanente comunicación entre los miembros participantes en los Comités Técnicos de Normalización y, en segundo lugar, porque las redes sociales han pasado a formar parte de la propia estructura de funcionamiento de los sistemas de gestión de calidad.

Aquellos que obtuvimos nuestras acreditaciones como Auditores ISO 9001 o nos especializamos en la norma ISO 17025 a comienzos del año 2000, todavía recordamos la ansiedad con la que esperábamos la modificación de la norma que permitía equiparar el archivo de registros en formato electrónico a los archivados en formato papel como medio perfectamente seguro para la custodia de este tipo de información. Hoy por hoy, la proliferación de escenarios virtuales, incluyendo la irrupción de la nube como espacio remoto de almacenamiento y procesamiento de la información, han convertido en necesaria la adaptación de todo el repertorio de normas de calidad a las crecientes posibilidades que ofrece este tipo de tecnologías de la información y las comunicaciones.

A partir de la experiencia adquirida en el seno del Instituto Nacional de Tecnologías de la Comunicación (INTECO), que desde su creación en el año 2006 ha impulsado decididamente la ejecución de políticas públicas en España destinadas a afianzar nuestra pujante industria del software, podemos advertir una doble tendencia en las líneas de desarrollo de la estandarización en el contexto de esta nueva sociedad de la información caracterizada por las redes 2.0 y 3.0, definibles por un creciente carácter interactivo y participativo.

La comunidad profesional de la industria 'software' y la proliferación de estándares de calidad

La necesidad de regular estándares que definan los elementos de calidad necesarios para promover el uso confiable y satisfactorio de las redes de comunicación ha llevado tempranamente a impulsar numerosos proyectos normalizadores. No obstante, el desarrollo de estándares en una comunidad tan dinámica, donde Internet facilita enormemente la elaboración y comunicación de propuestas, ha supuesto en ocasiones frustrados desarrollos de marcos normativos.

Podemos ilustrar ejemplos de estas dinámicas de aceleración en el desarrollo de estándares relativos a los cuatro espacios esenciales de normalización, como son la regulación del acceso a Internet, la regulación de la industria de desarrollo del software, la gestión de servicios TI o la seguridad de la información y las redes de comunicación. En este artículo nos centramos en los dos primeros ámbitos como son la accesibilidad y la calidad en la producción y el desarrollo del software.

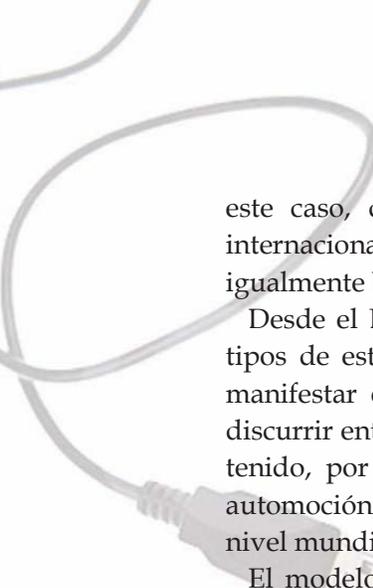
En primer lugar, debemos mencionar el grupo de estándares de accesibilidad web, que permiten establecer un diseño de los entornos virtuales en Internet de acuerdo con características accesibles y usables. La accesibilidad de las páginas web trata de aspectos relacionados con la codificación, el diseño y la presentación de información, dirigidos a permitir que las personas con algún tipo de discapacidad puedan percibir, entender, navegar e interactuar de forma efectiva con la web, así como crear y aportar contenido. La labor desarrollada por el World Wide Web Consortium (conocido por sus siglas W3C), permitió tempranamente disponer de las normas WCAG 1.0 y más recientemente WCAG 2.0. En España tempranamente se dispuso de una norma técnica que traducía al castellano las primeras, la UNE 139803:2004.

El hecho de anticipar una norma española como la UNE 139803 resultó esencial para que el legislador impulsara en el sector la necesidad de diseñar entornos web accesibles. La obligación de accesibilidad está regulada para las páginas de Internet (esto afecta a los elementos navegables que aparecen en las mismas, como vídeos, documentos pdf, etc.) de las Administraciones Públicas (Real Decreto 1494/2007, de 12 de noviembre), y se extiende también a las páginas web financiadas con fondos públicos y a las de entidades y empresas que se encargan de gestionar servicios públicos, en especial, de los que tengan carácter educativo, sanitario y de servicios sociales (Ley 56/2007, de 28 de diciembre).

En cualquier caso, el impulso a este marco normativo ha resultado esencial para mejorar la calidad de los servicios virtuales, ya que para las entidades públicas y empresas contar con páginas web accesibles es, aparte de una obligación legal, una ventaja competitiva. Además de poder ser utilizadas por las personas con discapacidad y las personas de edad avanzada, las páginas web accesibles tienen muchas otras ventajas, entra las que destacan las siguientes:

- Son más fáciles de utilizar por todos.
- Se adaptan mejor a los requerimientos de los nuevos dispositivos de acceso, como PDA, teléfonos móviles o televisión digital.
- Son mejor indexadas por los motores de búsqueda.
- Tienen menos costes de mantenimiento que las no accesibles.

Esta secuencia de promoción de estándares al margen de los grandes organismos de normalización se ha venido reproduciendo como una constante. Y así sucede en el segundo ámbito de normalización del sector como es el de la producción o desarrollo de software. En



este caso, dos estándares se han disputado la hegemonía en la industria del software internacional: el modelo generación y madurez CMMI y la conocida norma SPICE o ISO 15504, igualmente basada en este tipo de modelos de madurez.

Desde el Laboratorio Nacional de Calidad del Software de INTECO se impulsaron ambos tipos de estándar en el marco de la lógica neutralidad que los organismos públicos deben manifestar en torno a la elección de uno u otro tipo de estándar. La industria ha sabido discurrir entre uno y otro modelo en función de sus mercados preferentes. La norma SPICE ha tenido, por ejemplo, un notable predicamento en la industria del software destinada a la automoción, mientras que CMMI se ha convertido en una enseña para factorías de software a nivel mundial.

El modelo CMMI es una marca registrada por el Software Engineering Institute (SEI), un instituto federal estadounidense de investigación y desarrollo, fundado por el Congreso de los Estados Unidos en 1984 para desarrollar modelos de evaluación y mejora en el desarrollo de software que dieran respuesta a los problemas que generaba al ejército estadounidense la programación e integración de los subsistemas de software en la construcción de complejos sistemas militares. Está financiado por el Departamento de Defensa de los Estados Unidos y es administrado por la Universidad Carnegie Mellon (<http://www.sei.cmu.edu/>). El SEI alberga también el primer Centro de Respuesta a Incidentes de Seguridad (CERT), fundado por SEI en noviembre de 1988 por encargo de DARPA (Defense Advanced Research Projects Agency), para investigar y mejorar la seguridad de los sistemas de información del ejército y ejercer la coordinación en caso de emergencias (<http://www.cert.org/>). El fundador y primer director del SEI fue el profesor pamplonés Ángel Jordán. Esta doble relevancia le ha convertido en un prescriptor tecnológico de enorme prestigio mundial, pero sin duda el desarrollo de un esquema de acreditación de evaluadores internacionales que actúan como profesionales freelance, y la difusión libre de sus modelos de calidad, le han dado una gran ventaja en la difusión de este esquema de normalización de la mejora continua como modelo de gestión de los procesos de desarrollo de software.

Su éxito les ha llevado a promover la generación de estándares basados en su modelo de madurez en ámbitos como los Servicios TI (CCMI-SER) o la adquisición de software (CMMI-ACQ), entre otros.

Por su parte, dado que el tejido empresarial español está caracterizado por una notable presencia de las PYMEs, la difusión de las normas SPICE-ISO 15504 e ISO 12207: 2008 ha pretendido facilitar el acceso a la regularización de este sector emergente. Es de destacar que la ISO/IEC 12207: 2008 es una norma o estándar donde los procesos incluyen ingeniería de software y de sistemas, facilitando un acercamiento a los procesos del ciclo de vida del software, de modo que las factorías de software o pequeños grupos de desarrolladores logren realizar software de una forma sistemática más ágil, eficaz, eficiente y con una economía apropiada a sus recursos, para así poder competir en un mundo cada vez más globalizado.

La importancia de que estos modelos de calidad se hayan consolidado en las redes sociales integradas por los profesionales de ingeniería del software les ha conferido un carácter de sistemas líderes en la estandarización.

A modo de ejemplo del creciente éxito de este tipo de modelos en las redes profesionales y empresariales, en el año 2009 INTECO promocionó la primera traducción de la norma CMMI al castellano, continuada en 2012 por la traducción de la versión CMMI-DEV 1.3. Dichas versiones se han convertido en las más descargadas del portal del SEI tras la versión inglesa. En el caso de España, sus empresas encabezan ya el mayor número de empresas evaluadas en



Europa y ocupa la cuarta posición a escala mundial por detrás de Estados Unidos, China e India, según el informe del SEI publicado en marzo de 2012, “CMMI® for Development SCAMPISM Class A Appraisal Results 2011 End-Year Update”. El informe también señala que son 222 las organizaciones españolas que actualmente poseen el certificado CMMI, mientras que esta cifra era de menos de 10 en 2005, y ocupan unas posiciones muy destacadas países iberoamericanos como Brasil, con 143 organizaciones certificadas, México con 116 o Argentina con 70 entidades.

La nueva versión de la guía al castellano representa asimismo un ejemplo del potencial que representan las redes sociales. La coordinación de la traducción por el equipo docente de la Universidad Politécnica de Madrid, liderado por los profesores Jose A. Calvo, Gonzalo Cuevas y Tomás San Feliú, ha tenido en cuenta las contribuciones de diversos expertos latinoamericanos y españoles, tanto pertenecientes a la industria del software como al ámbito académico, intentando ambas partes responder a las crecientes necesidades y expectativas de los profesionales de la ingeniería del software.

Estos modelos de normalización responden, en definitiva, a una creciente demanda de traducciones de los estándares internacionales en materia de nuevas tecnologías, ante la necesidad de agilizar la incorporación de normas y métodos a los procesos de producción, que resulta clave a efectos de competir con mayor calidad, reduciendo a la vez los precios y plazos de entrega a los clientes. La internacionalización de los mercados conlleva la necesidad de superar las barreras lingüísticas, de modo que documentos tales como acuerdos, contratos, manuales y todo tipo de correspondencia comercial han de redactarse cada día en un mayor número de lenguas, y las comunidades y redes sociales se convierten en el foro que posibilita tanto la elaboración de normas en español como su rápida difusión y proliferación, convirtiendo a nuestra comunidad hispanohablante en la segunda más importante a nivel mundial en cuanto a estandarización.

La revolución social en las empresas

REDES SOCIALES / ARTÍCULO

La revolución social en las empresas

"Retos y oportunidades del beneficio del uso de las redes sociales"



César Franco
Responsable de
Desarrollo de Negocio
de Movilidad de HP
Vocal de Comisión de
Calidad del COIIM AIIM
Vocal del Comité
de CSTIC de la AEC

Cambios en el modelo de relación con los clientes

En todo el mundo y en todos los sectores, los clientes están cambiando. Están mejor informados y son cada vez más exigentes; se conectan desde cualquier ubicación, son cada vez más móviles y sorprendentemente sociales. Clientes particulares y profesionales adoptan las redes sociales a una velocidad acelerada; se conectan con otros, crean comunidades... y cambian rápidamente la propia naturaleza de sus relaciones con las empresas.

Los usuarios finales son más autodidactas al buscar productos y servicios. Se mantienen informados en un mundo que cambia rápidamente. Consultan los medios sociales en cualquier momento, en cualquier lugar y en una deslumbrante variedad de dispositivos. Las decisiones son a menudo instantáneas y las expectativas de los clientes, implacables.

Los clientes otrora leales ahora muestrean y se dirigen a nuevos lugares y productos. Una competencia agresiva emerge con facilidad con pocas barreras de entrada. Una imagen de marca —construida en décadas— puede ser seriamente dañada por una sensación viral durante la noche, ya sea verdadera o no. La realidad de los medios sociales afecta prácticamente a todos los sectores en todo el mundo. El ritmo de estos cambios se está acelerando, y las organizaciones están luchando para mantenerse al día. En este entorno, es necesario tener perspectiva sobre cómo las empresas pueden entender mejor, implementar y aprovechar las relaciones de los clientes en un mercado social emergente.

Retos y oportunidades

Con más de mil millones de personas utilizando las redes sociales y más

"Retos y oportunidades del beneficio del uso de las redes sociales"

Autor: César Franco Responsable de Desarrollo de Negocio de Movilidad de HP.

Vocal de Comisión de Calidad del COIIM AIIM.
Vocal del Comité de CSTIC de la AEC

Publicado en la Revista CALIDAD - N° IV • 2012

Cambios en el modelo de relación con los clientes

En todo el mundo y en todos los sectores, los clientes están cambiando. Están mejor informados y son cada vez más exigentes, se conectan desde cualquier ubicación, son cada vez más móviles y sorprendentemente sociales. Clientes particulares

y profesionales adoptan las redes sociales a una velocidad acelerada, se conectan con otros, crean comunidades... y cambian rápidamente la propia naturaleza de sus relaciones con las empresas.

Los usuarios finales son más autodidactas al buscar productos y servicios. Se mantienen informados en un mundo que cambia rápidamente. Consultan los medios sociales en cualquier momento, en cualquier lugar y en una deslumbrante variedad de dispositivos. Las decisiones son a menudo instantáneas y las expectativas de los clientes, implacables.

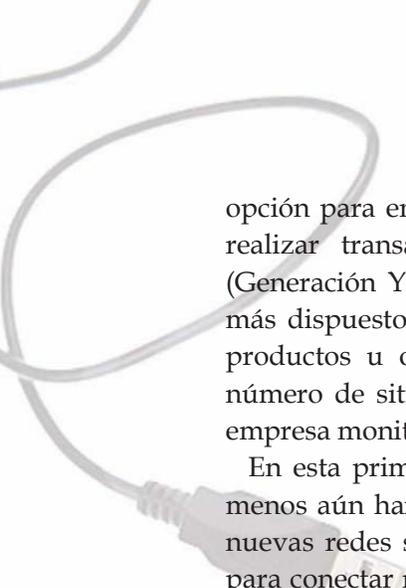
Los clientes otrora leales ahora muestrean y se dirigen a nuevos lugares y productos. Una competencia agresiva emerge con facilidad con pocas barreras de entrada. Una imagen de marca —construida en décadas— puede ser seriamente dañada por una sensación viral durante la noche, ya sea verdadera o no.

La realidad de los medios sociales afecta prácticamente a todos los sectores en todo el mundo. El ritmo de estos cambios se está acelerando, y las organizaciones están luchando para mantenerse al día. En este entorno, es necesario tener perspectiva sobre cómo las empresas pueden entender mejor, implementar y aprovechar las relaciones de los clientes en un mercado social emergente.

Retos y oportunidades

Con más de mil millones de personas utilizando las redes sociales y más consumidores uniéndose a diario, las empresas tienen dificultades para mantener el ritmo. Los consumidores confían cada vez más en foros, sitios de comparación de productos, servicios y precios, y otros medios de comunicación social al tomar decisiones de compra.

Mi generación (catalogada como Generación X) es propensa a conectarse como primera



opción para encontrar y evaluar productos y servicios, dar respuesta a problemas técnicos y realizar transacciones comerciales. Este efecto se acentúa en la siguiente generación (Generación Y) de consumidores. Familiarizados con los medios sociales, cada vez estamos más dispuestos a nombrar y a culpar a las marcas por problemas del servicio, fallos en los productos u otras experiencias negativas. Desafortunadamente, el volumen de tráfico, el número de sitios y el ritmo de los cambios hacen que sea extremadamente difícil para una empresa monitorizar y comprender qué dicen sus clientes en las redes sociales.

En esta primera etapa, pocas organizaciones disponen de una estrategia social definida, y menos aún han elaborado planes para beneficiarse proactivamente de las capacidades de las nuevas redes sociales. Es necesario entender la mejor manera de capacitar a sus empleados para conectar más directamente con los clientes. En este entorno, los sistemas de gestión de la relación con los clientes (CRM) y los procesos tradicionales siguen siendo importantes, pero insuficientes para las interacciones más dinámicas y complejas que ocurren en las redes sociales. En el nuevo contexto, las organizaciones necesitan un sistema de gestión de las relaciones sociales con los clientes (sCRM) para:

1. Monitorizar y seguir en las redes sociales cualquier impacto en la empresa.
2. Reducir los costes de CRM.
3. Adquirir, retener y fidelizar clientes.
4. Gestionar la marca, incluyendo el control de daños y la gestión de posibles crisis en los medios sociales.
5. Adquirir información de mercado de nuestros consumidores y de la competencia.

De esas actividades, la monitorización es actualmente el principal objetivo para muchas empresas. Realizan el seguimiento de la actividad en los medios sociales para medir el beneficio potencial de aproximarse a sus clientes a través de las redes sociales y ayudar a definir sus estrategias y presupuestos en medios de comunicación sociales. Obtienen conocimiento de rumores y sentimientos.

A continuación, las organizaciones aprovechan el sCRM para dar soporte a la gestión de marca: minimizar el impacto negativo de comentarios en los medios sociales, mejorar el servicio al cliente o derivar a los clientes hacia canales más tradicionales. Pueden ayudar a reducir los costes operativos, con una mejora de los foros de consumidores que alienten las interacciones entre ellos, generando focos de conocimiento, donde los propios clientes puedan compartir información técnica sobre productos y servicios.

Los medios sociales son una fuente de inteligencia de mercado, proporcionan información sobre las actitudes del consumidor, tendencias emergentes y actividades de la competencia. Aunque relativamente pocas organizaciones están utilizando sCRM para generar transacciones directas de venta, alternativas como el chat en línea pueden utilizarse para derivar las oportunidades a canales de venta más viables o gestionar mejor las consultas de los clientes.

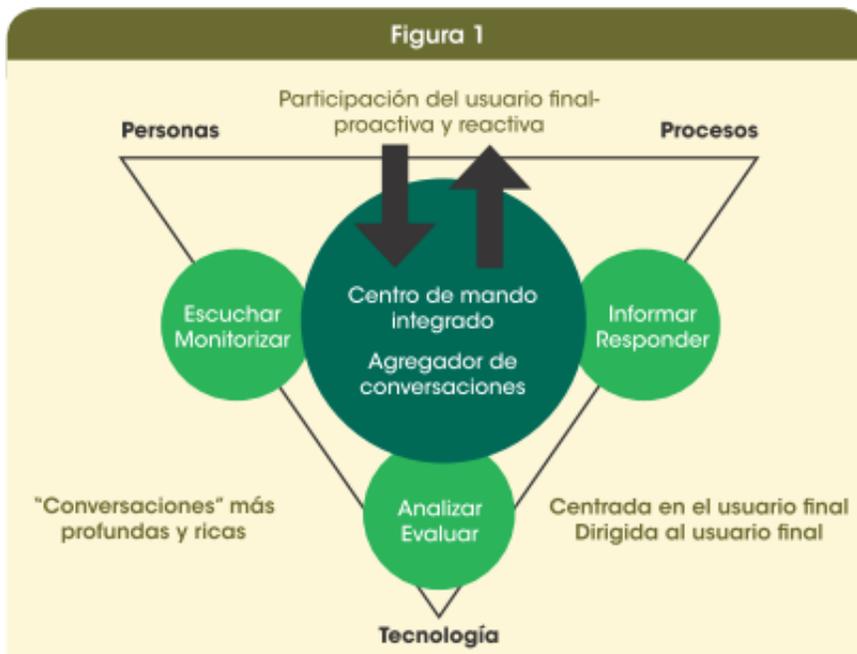
Escuchar, filtrar, actuar

Para obtener rentabilidad de una inversión en medios sociales, la empresa debe desplegar al mismo tiempo un modelo de gestión de la relación con el cliente que tenga en cuenta las redes sociales, y un caso de negocio sólido, alineado con los objetivos estratégicos de la organización.

Por ello, si queremos llegar a participar y gestionar conversaciones sociales entre clientes y empleados, es necesario escuchar y analizar antes de participar, y desplegar al tiempo un modelo de mejora continua para los procesos organizativos y de negocio y su orientación hacia las redes sociales.

Como se muestra en la figura 1, el modelo de HP centraliza las personas, procesos y tecnología necesarios para iniciar y organizar el diálogo con los clientes en medios sociales en lo que denominamos un Concentrador de Conversaciones. Las nuevas tecnologías permiten obtener información estadística de usuarios finales, sus opiniones dentro de sus redes, realizar el seguimiento de las mismas y facilitar la presentación de informes.

Esto implica ampliar el soporte dado a los usuarios a través de los canales tradicionales a los nuevos canales abiertos a través de la red social. Para ello, los equipos de soporte actuales necesitan disponer de guías a la toma de decisiones para la interacción del usuario a través todos esos canales. Deben estar basadas en reglas de negocio que consigan que la persona más adecuada



en la organización dé respuesta a las necesidades del cliente. El foco: resolver la necesidad del cliente en el primer contacto.

De la misma forma, las métricas e informes de los sistemas deben orientarse a valores como la resolución, satisfacción y valor entregado al cliente. La información analítica debe transformarse en variables orientadas al usuario: patrones, experiencia y comunidades. Las comunicaciones de marketing, promociones y segmentación se centran más en dichos modelos de segmentación que en las características técnicas de un producto o servicio.

Dentro de la organización, esto implica evolucionar los equipos humanos e infraestructura existentes integrando la información y procesos provenientes de medios sociales. Estos cambios se centran principalmente en dar a los propios usuarios mejor acceso al conocimiento y las herramientas necesarias para satisfacer sus requisitos de compra o soporte:

Impacto en la tecnología

- Monitorización automatizada de medios sociales.
- Capacidades analíticas: identificar patrones, tendencias, problemas y oportunidades.
- Información de presencia y distribución de la carga de trabajo para identificar y utilizar el recurso más adecuado a cada conversación.
- Alertas y análisis de indicadores que soporten una presencia más dinámica, ofertas, campañas y escaladas.
- Aplicaciones móviles y servicios geolocalizados que proporcionan valor en tiempo real a los usuarios finales.

Impacto en los agentes

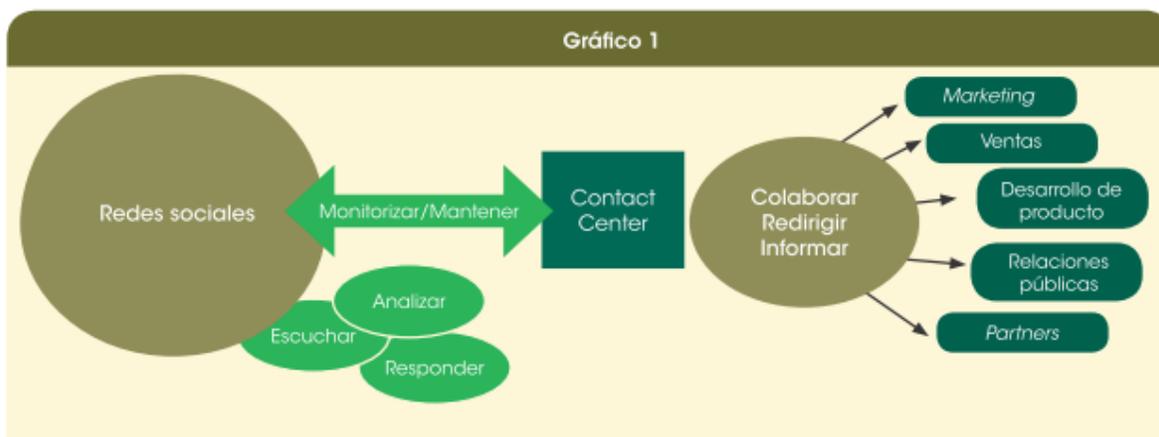
- Habilidades específicas, experiencia, idiomas y conocimientos culturales necesarios para realizar conversaciones más complejas con los clientes.
- Capacidad para presentar información basada en segmentación, datos históricos y perfiles de usuario para permitir diálogos más ricos e intensos con los usuarios.

Impacto en los procesos

- Reglas de toma de decisiones de negocio que permitan delegar autoridad al empleado en el primer punto de contacto.
- Sólidas capacidades de gestión del cambio.
- Modelos de contacto virtualizados, especialistas que pueden conectarse en cualquier lugar y otros recursos relevantes.

Es necesario desplegar una solución escalable y rentable para ver y captar las sensaciones de los clientes de una gran variedad de fuentes. Cada llamada al centro de asistencia, respuesta a una encuesta, contacto por móvil o correo electrónico, chat o un registro de actividad en la web o en redes sociales como Facebook puede ser indexada para localizar conceptos clave y relaciones.

Esta información permite a las empresas detectar y tomar medidas sobre nuevas tendencias, hacer frente a los problemas antes de que se conviertan en graves y amplificar las historias de éxito de clientes.

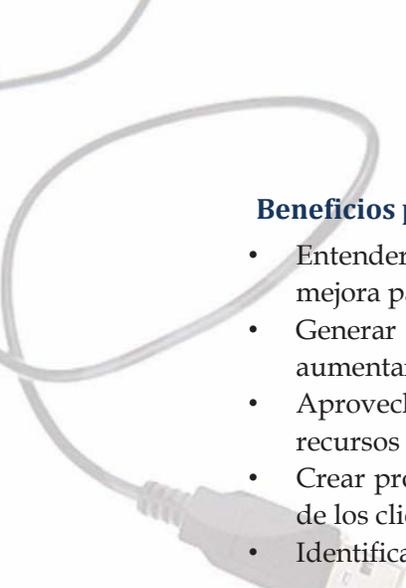


Enlazar el centro de atención a usuarios con las redes sociales

En todos los sectores y segmentos, una organización orientada al cliente puede aprovechar un CRM social para dos aspectos fundamentales:

Compromiso del cliente

- Convertir la información recogida de redes sociales en una mayor inteligencia del cliente.
- Establecer conversaciones más personales e íntimas con los clientes.
- Asegurar a los clientes una experiencia consistente a través de tiendas, puntos de contacto, sitios web y otros canales.
- Mejorar la relación con el cliente, el servicio, el soporte y su retención.
- Transformar la percepción y satisfacción del cliente en información accionable.



Beneficios para la empresa

- Entender y cuantificar la información de redes sociales para crear oportunidades de mejora para el negocio.
- Generar oportunidades adicionales de venta, venta cruzada y otras oportunidades de aumentar los ingresos.
- Aprovechar una comunicación proactiva con el cliente para desviar llamadas hacia recursos más útiles o eficientes en coste.
- Crear programas de marketing que se adapten mejor a las necesidades e intereses reales de los clientes.
- Identificar y abordar amenazas a la marca e imagen corporativas.

Modelo gradual de despliegue

HP ha diseñado una metodología que da soporte al despliegue gradual de las capacidades de la organización hacia los medios sociales. Se basa en un modelo de evaluación en cuatro niveles:

1. Análisis de sentimientos. La empresa realiza un seguimiento del mercado de los comentarios sobre su marca.
2. Defensa de la marca. La empresa dispone de recursos y procesos automatizados para responder y atraer a los consumidores que tienen problemas con la marca.
3. Resolución en el primer contacto. Resolver incidencias del servicio en los canales presencial y de autoservicio.
4. Servicio a ventas. Venta cruzada desde el servicio de soporte hacia la generación de una oportunidad de venta con ofertas muy concretas.



personas



procesos



tecnologías

Premios Calidad CSTIC

Los Premios **CALIDAD CSTIC** fueron instituidos por el Comité de Software de la AEC en este año 2009 con el objetivo de otorgar un reconocimiento público a aquellas empresas o instituciones que hayan utilizado con éxito la estrategia de la calidad en los sistemas y tecnologías de la información y las comunicaciones, o hayan contribuido de manera innovadora y decisiva a la creación o implantación de buenas prácticas en el ámbito de las TIC, para la mejora de la competitividad desde el punto de vista del negocio.

Pueden presentarse todas las empresas, así como otras instituciones públicas y privadas implantadas en España, que realicen y/o lideren algún trabajo o actividad ligada con las **tecnologías de la información y las comunicaciones**.

Las empresas y entidades que no sean miembros de la AEC deberán ser avaladas por un miembro del Comité de Software de la AEC para poder presentar su candidatura al premio.

Premio Calidad CSTIC 2009

En la edición 2009 de los Premios **CALIDAD CSTIC** tenían como objetivo dar un reconocimiento público a aquellas empresas o instituciones que hayan utilizado con éxito la estrategia de la calidad en los sistemas y tecnologías de la información y las comunicaciones, o hayan contribuido de manera innovadora y decisiva a la creación o implantación de buenas prácticas en el ámbito de las TIC, para la mejora de la competitividad desde el punto de vista del negocio.

Los premios se entregaron dentro del Congreso CSTIC 2009. Armando Veganzones, Presidente de la AEC, fue el encargado de entregar este galardón, en sus tres categorías.

Los premiados fueron:

PRIMER PREMIO

Endesa Servicios:

“Mapa de Procesos de Sistemas”.

Descripción del proyecto:

El proyecto MPRO consistió en la elaboración de un Mapa de Procesos común a toda la Dirección de Sistemas (D.S.):

1. Integrado, jerarquizado y simplificado.
2. Refleja los puntos clave de la relación entre los diversos actores.
3. Procesos homogéneos para representar una operativa común.
4. Orienta la Organización a la mejora continua de procesos.
5. Alineado con estándares internacionales (CMMI, ITIL).
6. Garante del cumplimiento de normativas comprometidas (Ley Sarbanes-Oxley, UNE 166002:2006).

La participación de las áreas afectadas durante la definición e implantación del modelo fue

clave para el éxito del proyecto.

El MPRO permite:

- Desarrollar y mantener el Software mediante una Metodología común.
- Verificar y evaluar toda la operativa de la D.S.
- Realizar un seguimiento global de una cartera de proyectos unificada.
- Realizar un seguimiento de los Procesos mediante indicadores del servicio ofrecido por proveedores y a clientes.
- Ser transformado mediante la Mejora Continua

MENCIONES ESPECIALES:

INTECO:

“Creación de un Laboratorio Nacional de Calidad del Software”

Descripción del proyecto:

El Laboratorio Nacional de Calidad del Software, en adelante LNCS, como parte del Instituto Nacional de Tecnologías de la Comunicación (INTECO), nace con el claro propósito de responder a diversas necesidades de la industria española. El proyecto LNCS tiene el objetivo global de convertirse en un observatorio tecnológico del sector de la calidad del software. Se trata de un proyecto pionero e innovador que apuesta por la creación e implantación de buenas prácticas en el ámbito de las TIC, cuyo objetivo es la mejora de la competitividad desde el punto de vista del negocio.

La calidad del software se fundamenta en tres conceptos clave: personas, procesos y herramientas. Enfocado a estos conceptos y atendiendo a su misión, el LNCS ha diseñado un marco de actuación que orienta todos los retos, necesidades y demandas de la industria española del software, con especial atención a las PYMEs. Líneas de actuación: 1.Sensibilización, difusión y formación. 2.Herramientas y servicios.

Vodafone España:

“Implantación en un Área de Tecnología de un Modelo de Calidad de Servicio basado en Experiencia y Satisfacción del cliente”.

Descripción del proyecto:

Dentro de las áreas en las empresas de telecomunicaciones, tradicionalmente Tecnología ha sido un área alejada de la interacción directa con el cliente.

Durante años el acercamiento de Tecnología al cliente ha sido a través de indicadores globales de satisfacción y de incidencias recogidas en los call center (aspectos alejados de la operativa del propio área), y de la mejora de indicadores internos tanto de proceso como de calidad (no siempre correlacionados con la percepción del cliente).

El proyecto que hemos llevado a cabo dentro del área de Tecnología de Vodafone España ha sido posible a través de la combinación de varias disciplinas como herramientas de geomarketing, análisis de investigación de mercado y modelos de calidad técnica, todos ellos englobados en una perspectiva de Satisfacción del Cliente.

Los resultados, además de culturales en la orientación del área, se han materializado en una situación de mejora competitiva, de acuerdo con la estrategia de la compañía.

Premios Calidad CSTIC 2011

El Comité de Calidad en los Sistemas y las tecnologías de la Información y las Comunicaciones **CSTIC** convocó la edición 2011 de los **Premios Calidad CSTIC**, organizados por la Asociación Española para la Calidad (AEC) con el objetivo de reconocer públicamente a aquellas empresas o instituciones que hayan utilizado con éxito la estrategia de la Calidad en las TIC o hayan contribuido de manera innovadora y decisiva a la creación o implantación de buenas prácticas en el ámbito de las TIC, para la mejora de la competitividad.

Los premios Calidad CSTIC se convocaron en dos categorías:

1. **Mejores prácticas en la prestación de servicios TIC**, en el proceso de desarrollo de software, en la gestión de la seguridad de la información y en el buen gobierno de las TIC.
2. **Mejora del negocio apoyada en el uso de las TIC**.

PREMIADOS

En la modalidad de “**Mejores prácticas en la prestación de servicios TIC**”:

ATOS SPAIN con el proyecto "**Kanban Software- Lean aplidado al mantenimiento del software**".



Resumen del proyecto: Dentro de un marco de mejora continua, la Línea de Servicios de Integración de Sistemas (LSIS) de Atos Origin SAE ha desarrollado un conjunto de iniciativas internas cuyo objetivo principal es el estudio y aplicación de metodologías y prácticas ágiles en la ejecución de proyectos en los que participa.

Estas iniciativas tratan de ejecutar algunas partes del mismo mediante la adopción de técnicas que permitan: Simplificar las tareas de gestión;

Desarrollar software que realmente aporte el valor que el cliente espera; Adaptarse a las necesidades cambiantes del cliente; Aumentar la motivación de los integrantes del equipo del proyecto; Reducir la no-calidad

Uno de los resultados de estas iniciativas ha sido Kanban Software, consistente en la aplicación de esta técnica Lean en el mantenimiento de software.

El principio de Kanban se basa en un sistema de información para controlar el estado y tiempo necesario de cada una de las tareas que tiene lugar en el mantenimiento de software.

En la modalidad de "Mejora del negocio apoyada en el uso de las TIC":
INDRA SOFTWARE LABS con el proyecto "IGEA más cerca del Cliente"



Resumen del proyecto: IGEA es una colección de componentes Software configurables que facilitan la optimización de la gestión de redes y activos de las empresas de Utilities, Defensa, Infraestructuras, Transporte, Telecomunicaciones, Seguridad y Medio Ambiente. Permite la representación gráfica y geográfica de entidades desde distintas fuentes de datos en una única vista, en modelos 2D y 3D. IGEA permite la representación de miles de objetos en tiempo real. Desde 2007, siguiendo CMMi L4, se ha definido un modelo para el desarrollo de Sw, lo

que implica mejor calidad de los desarrollos, menor tiempo de resolución de incidencias y menos errores del producto implantado. Se implanta CMMi L5 para lograr mayor seguridad y fiabilidad del Sw, mejor percepción del cliente, menor nº errores, mayor productividad en mantenimiento, orientando los esfuerzos a innovar el producto, fortaleciendo la mejora continua y orientándola a los beneficios de clientes y los objetivos de negocio. Los clientes reciben un servicio predecible.

Premios Calidad CSTIC 2013

La entrega de los premios la realizaron Pedro Iglesias, vicepresidente de la AEC y Ramiro Carballo Presidente del comité CSTIC, dentro del evento [CSTIC 2013](#).

En esta edición se presentaron 22 proyectos de distintas entidades públicas y privadas de toda España. a los premios repartidos entre las dos categorías quedado los siguientes resultados:

Categoría: Mejores prácticas en la prestación de servicios TIC

En la categoría de Mejores prácticas en la prestación de servicios TIC, en el proceso de desarrollo de software, en la gestión de la seguridad de la información y en el buen gobierno de las TIC, hubo cinco menciones a las empresas Alhambra-Eidos, Babel Sistemas de Información, BULL España SA, Gneis Global Services y Ministerio de Defensa/DIGENIN/Subdirección General de Tecnologías de la Información y Comunicaciones.



Y tres finalistas: EFRON Consulting, T-Systems ITC Iberia, S.A.U., Panel Sistemas Informáticos, S.L.



Resultando finalmente ganador **Panel Sistemas Informáticos** por su proyecto **La Excelencia Tecnológica a través de la Calidad**.



Resumen del proyecto: Desde nuestros inicios, la Excelencia Tecnológica y la Calidad forman parte de nuestro ADN, lo que nos llevó en 2011 a acreditarnos a la vez en el modelo CMMi – DEV y la norma ISO 15504, ambas en nivel 3 de madurez. Supuso una experiencia pionera, y Panel fue de las pocas empresas españolas que cumplían con estas normativas por separado en el nivel de madurez 3, y la primera empresa europea que las cumplía al mismo tiempo.

Nuestra metodología, llamada OGMA, es hoy la evidencia del compromiso de nuestra Organización con una eficaz prestación de los servicios y la mejora continua de los procesos, y es utilizada en las 3 Factorías Software que Panel tiene en España (Madrid y Ciudad Real) e India (Ahmedabad). En 2013, tras los éxitos alcanzados, hemos decidido dar un paso más hacia la mejora continua y la calidad, iniciando un proceso de evaluación en el modelo CMMi Servicios nivel 2 de madurez, para la prestación de servicios de pruebas e implantación desde nuestra Testing Factory.

Categoría: Mejora del negocio apoyada en el uso de las TIC

En la categoría Mejora del negocio apoyada en el uso de las TIC se realizó una mención a MakeSoft Technologies



Y quedaron dos finalistas, decidiendo el jurado que el Premio fuera compartido por ambos organismos.

- **WEETSI: el portal Web de Formación, Empleo y empresa del Ayuntamiento de Vélez-Málaga** del Organismo Autónomo Local de Desarrollo Integral del Municipio.



Resumen del proyecto: La Web de Servicios de Información Telemáticos de Formación, Empleo y Empresa del Ayuntamiento de Vélez-Málaga (WeetSI) supone un salto cualitativo en el acceso de los ciudadanos a los servicios relacionados con la formación, el empleo y el ámbito de la empresa. El alcance de Internet y de las Tecnologías de la Información y la Comunicación (TIC) en nuestros días, supone una herramienta efectiva para ofrecer el máximo número de servicios de información y tramitación a los ciudadanos en su relación con la Administración Pública. El alcance de WeetSI como proyecto de mejora de los servicios de empleabilidad del Municipio de Vélez-Málaga, supone la integración de información en un espacio web unificado, con un alto nivel de usabilidad, facilitador de la participación e interacción del ciudadano y con el firme propósito de agilizar los trámites en materia de formación, empleo y empresa. Enlace al portal: weetsi.velezmalaga.es

- **Herramienta TIC de Gestión de Licencias Urbanísticas del Ayuntamiento de Madrid** (Área de Gobierno de Urbanismo y vivienda).



Resumen del proyecto: El Ayuntamiento adoptó medidas de mejora del servicio de licencias urbanísticas. Una fue la necesidad de una Herramienta de apoyo a la gestión de Licencias Urbanísticas, con el objetivo de que, de forma estructurada, accesible y utilizando TIC, se delimitara y favoreciera el conocimiento de la normativa que es de aplicación y se alcanzara una mejora de eficacia y eficiencia, homogeneizando los criterios de su utilización, tanto en la gestión interna de la Administración como de cara a los particulares solicitantes de licencias. La Herramienta ofrece a los servicios municipales una información omnicomprendensiva de los aspectos sistemáticos, normativos e interpretativos que afectan a la gestión y tramitación de licencias urbanísticas (normas, instrucciones,...), y a los sujetos externos, específicamente, un marco de referencia objetivo y fiable para la formalización de sus solicitudes. Adquiere especial dimensión como guía a particulares en la figura de la declaración responsable.

Empresas y miembros del comité

- **Presidente:**

- Ramiro Carballo Gutiérrez [Caelum Information & Quality Technologies, S.L.](#)

- **Vicepresidentes:**

- Antonio Moya Catena
- Domingo Manuel Gaitero Gordillo [Proceso Social](#)
- José Luis Báez García [FCC Servicios Industriales y Energéticos, S.A](#)
- María Teresa Atienza García [INDRA Software Labs, S.L.](#)

- **Secretario:**

- Juan Antonio Caloto Caloto [AEC Asociación Española para la Calidad.](#)

- **Vocales:**

- Ana Isabel González Fernández
- Carmen De León Molina [IBERDROLA, S.A.](#)
- Cesar Franco Ramos [Colegio Oficial de Ingenieros Industriales de Madrid](#)
- Cristina Fernández Fernández
- Felix Antonio Barrio Juárez [INTECO \(Instituto Nacional De Tecnologías De La Comunicación\)](#)
- Fernando Alonso Ruenes [Telvent Servicios Compartidos](#)
- Francisco Javier Casquero García [GTD Sistemas de Información, S.A.](#)
- Inocencia García Martín
- José Ángel Valderrama Antón [AENOR \(Asociación Española De Normalización Y Certificación\)](#)
- José Bautista Barato Arroyo [PMPEOPLE](#)
- José Ignacio Conde Arias-Camisón [IDICE Investment, Development & Innovation Consulting for Enterprises, S.L.](#)
- Josefina Alonso Nocelo
- Juan Antonio López Jiménez [GTD Sistemas de Información, S.A.](#)
- Juan Bautista Pérez Mínguez [Mº de Defensa- \(Subd. Gral. Serv. Tec. y Telecom.\)](#)
- Juan Carlos Vigo [ATI](#)
- Julio J. Ballesteros [Quint Wellington Redwood Group, S.L.](#)
- Laura García Borgeñón [Instituto Tecnológico de Aragón](#)
- Luis Castillo Zugasti [Deimos Space, S.L.U.](#)
- Manuel Arturo Lea Pereira [GMV Aerospace and Defence, S.A.](#)
- Miguel García-Menéndez
- Mikel Emaldi Abasolo Tecnalia - ICT división - [European Software Institute \(ESI\)](#)
- Raquel Martín Gómez [Mº de Defensa- \(Subd. Gral. Serv. Tec. y Telecom.\)](#)
- Rubén Romero Bousoño [ISDEFE Ingeniería De Sistemas Para La Defensa De España, S.A.](#)
- Santiago González Lobo [MTP Software Quality Assurance](#)
- Víctor Hervías Delgado [BCM Business Continuity Management](#)



personas



procesos



tecnologías



Publicación financiada por:

Proceso2ocial
Pasión por la tecnología

www.procesosocial.com

Documento conmemorativo del 30 aniversario del
Comité de Calidad en los Sistemas y Tecnologías
de la Información y la Comunicación

Financiado por:

Proceso2ocial

Pasión por la tecnología

www.procesosocial.com

Asociación Española para la Calidad
Claudio Coello, 92
28006 Madrid
aec@aec.es
www.aec.es