

INTRODUCCIÓN

La mayor importancia de las nuevas tecnologías de la información y su creciente presencia en los diversos ámbitos de la industria moderna (robots, centros de control, etc.) y sus productos finales (aviación, automóviles, electrodomésticos, telefonía, etc.) conlleva cada vez más la presencia de programas informáticos que gobiernan muchas de sus prestaciones, o bien como herramientas que el cliente empleará en su propio beneficio.

Basta con observar la frenética actividad que ha supuesto para la industria, las empresas de servicios y la Administración el enfrentarse al tan temido "efecto 2000", así como el gasto que ha conllevado la revisión y modificación de los programas, para vislumbrar la punta de un iceberg: la falta de un control riguroso y sistemático de la calidad del software de gestión. En el presente trabajo se aborda este tema que ayudaría a proveedores y clientes desde el comienzo del diseño de una aplicación específica de software para su negocio o actividad.

Algunos antecedentes al concepto de calidad.

A lo largo de toda la historia la búsqueda y el afán de perfección por parte del hombre ha sido constante, de tal forma, que el interés por el trabajo bien hecho y la necesidad de asumir responsabilidades sobre la labor efectuada poco a poco derivó en el concepto de calidad.

Con la revolución industrial comienza a desaparecer el artesanado, se crean grandes organizaciones y los antiguos artesanos se transforman en los trabajadores de las empresas. En esta época Taylor elaboró su teoría acerca de la "gestión científica del trabajo", cuyo objeto fue la preparación de normas para que los trabajadores las cumplieren. Comenzó con ello la instauración paulatina de la división del trabajo, lo que suponía que los operarios interviniesen solamente en algunas operaciones del proceso productivo. Este hecho provocó la necesidad de que surgiese la figura de los empleados dedicados a tareas de inspección, aunque se prestaba más atención a la forma de realizar el trabajo (los procesos) que a la calidad de los productos.

Finalmente el control de calidad moderno o control de calidad estadístico comenzó en los años 30 del siglo XX con la aplicación industrial del cuadro de control ideado por el Dr. W. A. Shewhart, de Bell Laboratories, que fue el inventor de los conocidos gráficos de control.

Continuando en este proceso cronológico destaca el hecho de que pasados unos años del final de la II Guerra Mundial los japoneses comienzan a hacer verdadero énfasis en la calidad. En 1950 la Unión de Científicos e Ingenieros Japoneses realizó un seminario cuyo conferenciante, el Dr. W.E. Deming, desarrolló los siguientes temas: Cómo mejorar la calidad mediante el ciclo de planear, hacer, verificar y actuar; La importancia de captar la dispersión en las estadísticas; Control de procesos mediante el empleo de cuadros de control y su aplicación.

La visita de Juran marcó una transición en las actividades de control de calidad y creó un ambiente en que se reconoció el Control de Calidad como un instrumento de gerencia abriéndose las puertas para el establecimiento del control total de calidad tal como se concibe hoy.

Calidad: definiciones.

Una vez revisados los antecedentes del concepto calidad para concretar su significado se van a reproducir tres definiciones de calidad emanadas de personas y entidades de reconocido prestigio:

- "Conjunto de esfuerzos efectivos de los diferentes grupos de una organización para la integración del

desarrollo, del mantenimiento y de la superación de la calidad de un producto, con el fin de hacer posible la fabricación y servicio a satisfacción completa del consumidor y al nivel más económico" [Feigenbaun, Deming y Juran]

- "La mejor calidad que una empresa puede producir con su tecnología de producción y capacidades de proceso actuales, y que satisfará las necesidades de los clientes, en función de factores tales como el coste y el uso previsto" [Dr. Kaoru Ishikawa]
- "La gestión de calidad en la empresa es el proceso de identificar, aceptar, satisfacer y superar constantemente las expectativas y necesidades de todos los colectivos humanos relacionados con ella, clientes, empleados, directivos, propietarios, proveedores y la comunidad con respecto a los productos y servicios que esta proporciona" [consultora Arthur Andersen]

De todas estas definiciones se extraen una serie de parámetros básicos que definen la calidad: si se desea producir productos y servicios de buena calidad para el consumidor será necesario decidir por adelantado que calidad de producto (o servicio) planificar (calidad de diseño), producir (calidad de fabricación) y vender (calidad que desea el cliente).

Calidad del software.

A la hora de definir la calidad del software se debe diferenciar entre la calidad del producto software y la calidad del proceso de desarrollo de éste (calidad de diseño y fabricación). No obstante, las metas que se establezcan para la calidad del producto van a determinar los objetivos a establecer de calidad del proceso de desarrollo, ya que la calidad del primero va a depender, entre otros aspectos, de ésta. Sin un buen proceso de desarrollo es casi imposible obtener un buen producto. Este proceso constituye el objeto del presente trabajo.

Pero la calidad del producto software se diferencia de la calidad de otros productos de fabricación industrial, ya que el software tiene sus propias características específicas:

- El software es un producto mental, no restringido por las leyes de la Física o por los límites de los procesos de fabricación. Es algo abstracto, un intangible.
- Se desarrolla, no se fabrica. El coste está fundamentalmente en el proceso de diseño, no en la posterior producción en serie, y los errores se introducen también en el diseño, no en la producción.
- Los costes del desarrollo de software se concentran en las tareas de Ingeniería, mientras que en la fabricación clásica los costes se acentúan más en las tareas de producción.
- El software no se deteriora con el tiempo. No es susceptible de los efectos del entorno y su curva de fallos es muy diferente de la del hardware. Todos los problemas que surjan durante el mantenimiento estaban allí desde el principio y afectan a todas las copias del mismo; no se generan nuevos errores.
- Es artesanal en gran medida. El software, en su mayoría, se construye a medida, en vez de ser construido ensamblando componentes existentes y ya probados, lo que dificulta aún más el control de su calidad.
- El mantenimiento del software es mucho más complejo que el mantenimiento del hardware. Cuando un componente del hardware se deteriora se sustituye por una pieza de repuesto, pero cada fallo en el software implica un error en el diseño o en el proceso mediante el cual se tradujo el diseño en código máquina ejecutable.
- Es engañosamente fácil realizar cambios sobre un producto software, pero los efectos de estos cambios se pueden propagar de forma explosiva e incontrolada.
- Como disciplina, el desarrollo de software es aún muy joven, por lo que las técnicas de las que dispone aún no están perfeccionadas.
- El software con errores no se rechaza. Se asume que es inevitable que el software presente algunos errores de poca importancia.

También es importante destacar que la calidad de un producto software debe ser considerada en todos sus estados de evolución (especificaciones, diseño, códigos,...). No basta con verificar la calidad del producto una

vez finalizado cuando los problemas de mala calidad ya no tienen solución o su reparación es muy costosa.

La problemática general a la que se enfrenta el software es:

- Aumento constante del tamaño y complejidad de los programas.
- Carácter dinámico e iterativo a lo largo de su ciclo de vida, es decir que los programas de software a lo largo de su vida cambian o evolucionan de una versión a otra para mejorar las prestaciones con respecto a las anteriores.
- Dificultad de conseguir productos totalmente depurados, ya que en ningún caso un programa será perfecto.
- Se dedican elevados recursos monetarios a su mantenimiento, debido a la dificultad que los proyectos de software entrañan y a la no normalización a la hora de realizar los proyectos.
- No suelen estar terminados en los plazos previstos, ni con los costes estipulados, ni cumpliendo los niveles deseables de los requisitos especificados por el usuario.
- Incrementos constantes de los costes de desarrollo debido entre otros, a unos niveles de productividad bajos.
- Los clientes tienen una alta dependencia de sus proveedores por ser en muchos casos aplicaciones a "medida".
- Procesos artesanales de producción con escasez de herramientas.
- Insuficientes procedimientos normalizados para estipular y evaluar la productividad, costes, y calidad.

Todo lo anterior puede concretarse en:

- Ausencia de especificaciones completas, coherentes y precisas previas por parte del cliente, así como posteriores por parte de los proveedores del software.
- Ausencia de la aplicación sistemática de métodos, procedimientos y normas de ingeniería del software.
- Escasez o ausencia de entornos integrados de programación.
- Escasez de uso de técnicas actuales y automatizadas para la gestión de proyectos.
- Escasez de personal con formación y experiencia en los nuevos métodos, normas y uso de entornos y utilidades de programación.
- Otros derivados del grado de desarrollo técnico y organizativo de cada compañía.

La calidad a través de la normalización en la ingeniería del software y su problemática.

La normalización consiste en un proceso donde se elaboran guías, normas y convenciones sobre una determinada materia, con el objeto de definir, simplificar y especificar las actividades relacionadas con la materia de que se trate.

La Ingeniería del Software (IS) se ha ido desarrollando en los últimos 15 años, a través de la creación e implantación en la industria software de métodos, procedimientos, técnicas y útiles que tratan de cubrir las necesidades de cada una de las etapas del ciclo de vida de un producto software, desde la definición de sus requisitos hasta su mantenimiento una vez el producto comience a emplearse. Y ello con las restricciones generales de todos los procesos modernos de ingeniería, esto es, la necesidad creciente de incrementar la productividad de la programación mejorando y garantizando, simultáneamente la calidad del producto resultante.

La creación e implantación de normas de desarrollo del software son un auténtico desafío que tiene la IS como medio de comunicación para transferir sus métodos, técnicas y procedimientos a la industria del software para el diseño y desarrollo de nuevos productos. Estas normas tienen como criterio general de desarrollo maximizar la comunicación entre los profesionales del software a través de la definición de documentos generales que se han de producir, proveyendo de guías que indican a nivel de detalle el contenido de dichos

documentos y recomendaciones de las actividades que hay que realizar durante todo el proceso de producción del software. En pocas palabras, las normas de IS son la solución a una de las mayores necesidades de la industria del software actual: la comunicación mas adecuada y precisa entre sus profesionales.

A medida que ha ido aumentando la necesidad de un software más fiable, se ha reconocido que las normas de ingeniería del software (NIS) son una contribución fundamental para asegurar la producción de software de calidad. Además una consecuencia del objetivo genérico de mejorar la comunicación es que se reducen los costes por un aumento de productividad y una mejora de la calidad de los desarrollos de software.

En relación a las normas los profesionales se encuentran con un problema fundamental: la dispersión de las normas relativas al software que, con frecuencia, han sido creadas por organismos muy diversos, bajo enfoques distintos y destinadas a ámbitos de actuación diferentes. Muchas compañías, por su parte, se han visto obligadas a generar sus propias normas cuando no disponen de unas de ámbito general. De hecho muchas organizaciones desarrollan sus propios conjuntos de normas adecuándolas a sus fines específicos. Pero también se dan casos en que organizaciones distintas tienen los mismos objetivos por lo que resultaría razonable su colaboración y, en todo caso, la adopción de las normas de la organización que tenga más avanzados sus desarrollos y un ámbito de actuación más amplio. Puede afirmarse que en la actualidad se ha llegado a un nivel de madurez en la industria del software que ha permitido a todos los implicados que exista un interés por aunar sus experiencias y esfuerzos para crear normas generales que abarquen sus áreas de interés.

Estos esfuerzos varían en cuanto al tipo de industrias o usuarios así como en lo relativo a los logros alcanzados, pero la tendencia actual es hacia la normalización del proceso de desarrollo software a través de normas que conduzcan a homogeneizar los planes de garantía de calidad de él, los planes de gestión de la configuración del software, la documentación de sus pruebas, etc.

En cuanto al proceso de creación e implantación de las normas, nos remitiremos a las dictadas por el IEEE: una nueva idea puede tenerla y sugerirla cualquiera de sus miembros, pasándola a continuación al Comité de Normas de IS, el cual cursará una Petición de Autorización de Proyecto (PAP) que se transmite a todos los grupos que hacen normas y al ANSI (American National Standard Institute) que se encargará de la coordinación.

La PAP define el objetivo, ámbito, descripción y principales contactos para llevar a cabo el proyecto. La etapa siguiente corresponde al desarrollo de la norma a cargo de un grupo de trabajo. Cualquiera puede adherirse al grupo y el IEEE se encargará de implicar a todos los interesados para que participen en reuniones con los demás miembros (unas cuatro reuniones anuales). También pueden intervenir a distancia detallando las contribuciones oportunas y remitiéndolas al grupo correspondiente. El proceso será más enriquecedor cuantos más profesionales de distinto nivel y más organismos intervengan.

Una vez terminado un desarrollo se concretará en un documento (borrador o draft) que pasará por varias revisiones como consecuencia del proceso de valoración y votación para aceptarlo. Durante este periodo habrán de votar al menos el 75% de los miembros del grupo de votación (unos 100 expertos) y a su vez el 75% debe ser positivo. Tras su aprobación se distribuirá la nueva norma. En cuanto al tiempo medio que se tarda desde que se inicia la elaboración de una norma hasta que se aprueba es de tres años. Finalmente, como trabajo complementario se procede difundir la norma a través de Seminarios.

La etapa siguiente en este ciclo corresponde con la de uso de la norma por las empresas u organismos. Durante esta fase los usuarios proporcionan comentarios positivos o negativos acerca de la norma, según su propia experiencia, y en ocasiones aceptan o modifican las normas para sus fines específicos. Las modificaciones se remiten al consejo de normas del IEEE para su evaluación por el grupo de trabajo entrando en una nueva fase del ciclo de elaboración de normas. Finalmente se decidirá dejarla igual, modificarla o eliminarla si ya ha sido superada por una nueva norma. El ciclo completo dura cinco años.

Los modelos de calidad del software de gestión.

En general los modelos de calidad definen a ésta de forma jerárquica, o sea la calidad se produce como consecuencia de la evaluación de un conjunto de indicadores o métricas en diferentes etapas:

En el nivel más alto de la jerarquía se encuentran los factores de calidad definidos a partir de la visión del usuario del software, y conocidos también como atributos de calidad externos.

Cada uno de los factores se descompone en un conjunto de criterios de calidad, o sea aquéllos atributos que cuando están presentes contribuyen a obtener un software de la calidad. Se trata de una visión de la calidad técnica, desde el punto de vista del producto software y se les denomina también atributos de calidad internos.

Finalmente para cada uno de los criterios de calidad se definen un conjunto de métricas o medidas cuantitativas de ciertas características del producto que indican el grado en que dicho producto posee un determinado atributo de calidad.

De esta manera, a través de un modelo de calidad se concretan los aspectos relacionados con ella de tal manera que se puede definir, medir y planificar. Además el empleo de un modelo de calidad permite comprender las relaciones que existen entre diferentes características de un producto software.

En contra de los modelos de calidad pesa que aún no ha quedado demostrada la validez absoluta de ninguno de ellos.

CONCLUSIÓN

Como se ha visto a lo largo de esta exposición, hoy día se comienza a imponer la obligación de normas de calidad del software donde un fallo en la información, o en el tratamiento de ésta puede llevar a fallos catastróficos y de consecuencias imprevisibles. Por ello las organizaciones están exigiendo controles de calidad más rigurosos en la confección de su software.

Hoy día el tener implantados sistemas de calidad en la empresa, debe llevar no solo el tener que instalar la metodología del sistema de calidad sino también sistemas de información que controlen y coordinen el sistema, sistemas automáticos, sistemas documentales, etc. Por todo ello la implantación de sistemas de calidad en cualquier empresa u organización debe implicar que también el software que empleen los posea, y ello repercute en la obligación de que sus proveedores de software los hayan empeado en la elaboración de sus productos. De esa forma se evitarían defectos provenientes de los sistemas de información.

BIBLIOGRAFÍA

Kitchenham, B.; Towards a Constructive Quality Model, Software Engineering Journal, Vol.2, N. 4, pp. 105-113, 1987.

Miller, E., Howden, W. E., Tutorial, Software Testing & Validation Techniques, 2a ed., IEEE Computer Society Press, 1981.

Murine, G.E. , Integrating software quality metrics with software QA, Quality Progress vol.21, no.11; pp. 38-43; Nov. 1988.

Oman, P. W. , A case study in SQA Audits, Software Quality Journal nº 2, pp. 13-27, 1993.