

Implementación de Software en las Empresas

Implementation of Software in Companies

Andrés Roldán, Ingeniero de Sistemas,
Universidad Incca de Colombia, Colombia,
ing.andres26@hotmail.com

Pablo Gómez, Ingeniero Industrial,
Universidad Distrital Francisco José de
Caldas, Colombia,
pgomez35@hotmail.com

Jonathan Correa, Ingeniero de Sistemas,
Universidad Incca de Colombia, Colombia,
jonathan2345@gmail.com

Artículo de Investigación

Para citar este artículo:

Martínez, A., Correa, J., & Gómez, P. (2016). Implementación de modelos de software en las empresas. *Tecnología Investigación y Academia*,. <http://revistas.udistrital.edu.co/ojs/index.php/tia/article/view/4886> 3(2), 1-8

Fecha de recepción: 21-01-2015

Fecha de aceptación: 02-12-2015

Resumen

Aunque se sabe que los conocimientos de los modelos de software y la aplicación de ellos es un factor fundamental para obtener productos de calidad definiendo un proceso para su consecución, se pudo establecer a partir de una serie de encuestas que las empresas de software y sus ingenieros pueden tener los conceptos pero no saben en qué consiste la implementación de dichos modelos; no saben realmente si los utilizan dentro de su compañía ni qué éxito tienen estos en los diferentes proyectos. Siendo esto uno de los factores iniciales por los cuales los proyectos de software fracasan.

Palabras clave: modelos de cuarta generación, modelos de software, modelos DRA, modelos evolutivos, casa de software, ingeniero de desarrollo, líder técnico.

Abstract

Although it is known that the knowledge of software models and their implementation is a key factor to obtain quality products defining a process for their achievement, it could be established from a series of surveys that software companies and their engineers can have the concepts but they don't know what the implementation of these models is; they don't really know if they use these within their company and how successful have these in different projects. This is one of the initial factors why software projects fail.

Keywords: fourth generation models, software models, RAD models, evolutionary models, software house, development engineer, technical leader.

I. INTRODUCCIÓN

En la actualidad se hace indispensable la utilización de los modelos de software como uno de los factores críticos que ayudan a la consecución de productos de calidad, determinando un modelo de acuerdo a la situación y al problema que resolverá. Es importante definir un modelo y proceso “porque proporciona estabilidad, control y organización a una actividad que puede, si no se controla, volverse caótica. “Según Pressman, R. (2002). [1]

Pero en el entorno real : ¿estos modelos son aplicados en los distintos proyectos de software en las compañías colombianas? Para ello se establece en este artículo el resultado obtenido de una serie de veinte encuestas realizadas a tres casas de software¹ y una de salud² encuestando a sus ingenieros.

Para la realización de la encuesta se trabajó la herramienta de GoogleDrive³, la cual permitió llegar a los ingenieros de forma rápida y efectiva. La encuesta se encuentra publicada en la siguiente dirección:

<https://docs.google.com/forms/d/1UkiMPdWaxz2wYAAiYt5junnBLIMQvKxArSh9lpAOxv4/viewform> y los resultados en la dirección:

<https://docs.google.com/forms/d/1UkiMPdWaxz2wYAAiYt5junnBLIMQvKxArSh9lpAOxv4/viewanalytics>

Se logra saber si realmente en las empresas aplican estos modelos y si sus ingenieros tienen claro los conceptos para su implementación, para así poder afirmar que uno de los factores fundamentales por los cuales los proyectos de software no son exitosos es a causa de la falta de conocimiento y aplicación de los modelos de software.

¹ Stefanini. Multinacional norteamericana reconocida a nivel mundial: <http://stefanini.com/es/> Intergupo. Multinacional latinoamericana reconocida en el país y una de las pocas que está certificada en CMMI nivel 5: <http://www.intergrupo.com/>

Componente Serviex. Multinacional chilena con tres años en el mercado colombiano: <http://www.componenteserviex.com/>

² Hospital San Ignacio. Fundado en el año 1942: <http://www.husi.org.co/>

³ <https://drive.google.com/>

Todo lo anterior desglosando las preguntas y respuestas obtenidas por parte de los ingenieros y comparándolas con los conceptos establecidos por los académicos y a su vez estableciendo si las casas de software están a la vanguardia en la implementación de los últimos modelos de software y qué cantidad de proyectos han sido exitosos aplicando dichos modelos.

II. CONTENIDO

1. No manejar un modelo es una opción

Se parte del siguiente hecho: “Las empresas tienen la opción de trabajar o no en base a un modelo de software.”[2]

Las empresas son libres de tomar la decisión que crean más adecuada pero la decisión de no tener un modelo de software implica un riesgo en la consecución de resultados, ya que desde la creación de la “Ingeniería de Software” se ha trabajado con base en la experiencia por la creación de modelos de software que ayuden a establecer un proceso efectivo para la consecución de productos de calidad.

Por lo tanto lo aconsejable es establecer un modelo que sea el más adecuado según la problemática y el entorno del proyecto.

2. Resultados de las encuestas

Inicialmente se diseñaron las preguntas y se publicaron en línea con la ayuda de Google, luego se invitó a algunos ingenieros a diligenciar las encuestas, posteriormente se estudió la información obtenida.

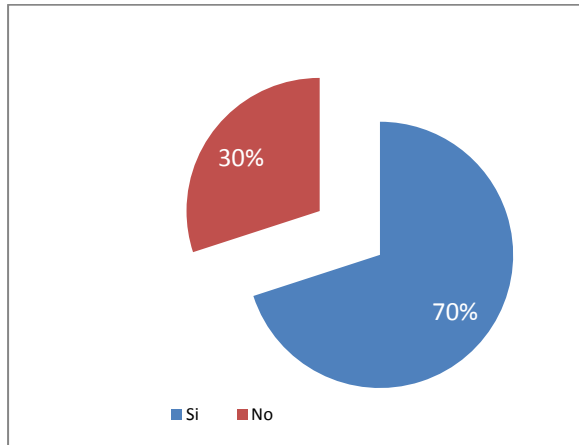
A continuación se mencionan las preguntas realizadas y los resultados establecidos:

Como se puede evidenciar en la tabla todos los encuestados hacen parte de proyectos de software dentro de sus compañías ya que es la forma más verídica de obtener la información necesaria para realización del presente artículo.

2.1 Preguntas de conocimiento general

¿Sabe qué es modelo evolutivo en el proceso de software?

Figura 1. Pregunta 1 ¿Sabe qué es el modelo evolutivo en el proceso de software?



La mayoría de los ingenieros encuestados dieron una respuesta afirmativa, pero al preguntarles en qué consiste se encontró que algunos no tienen un concepto claro acerca del modelo evolutivo, algunas respuestas fueron:

“Parte de un levantamiento de requerimientos superficial para que el resto del proceso pueda iniciar, este va evolucionando a medida que avanza el proceso de creación de software.”

“Creo que es el que llaman iterativo, en donde definen varias iteraciones, las cuales están divididas por las fases del modelo en cascada.”

“Es un modelo basado en procesos el cual al finalizar entra un producto o una evolución del software.”

La respuesta más acertada o más cercana a la realidad que se encontró fue la siguiente:

“Un modelo evolutivo es aquel que se elabora por medio de especificaciones que van siendo más específicas con la medida del tiempo, así mismo la evolución se mide dependiendo cómo es la experiencia con el sistema y las necesidades del cliente”

Definición esperada:

“El modelo evolutivo entrega el software en partes pequeñas, pero utilizables, llamadas

(evoluciones). En general, cada incremento se construye sobre aquél que ya ha sido entregado.”[1]

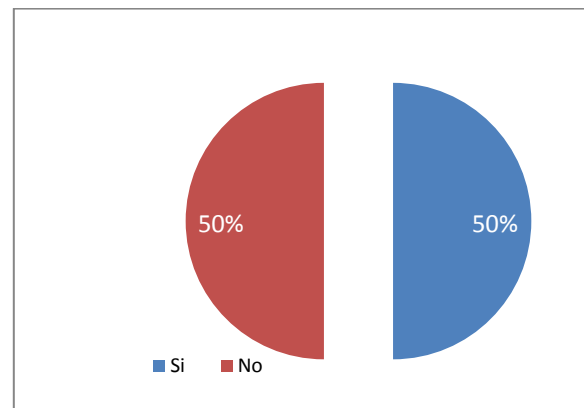
En esta pregunta se puede decir que la mayoría de los ingenieros tienen la noción de modelo evolutivo.

Ahora se mostrará qué es lo que ocurre cuando se pasa a otro modelo como el DRA.

Se realizó la siguiente pregunta:

¿Sabe qué es modelo DRA en el proceso de software?

Figura 2. Pregunta 2 ¿Sabe qué es modelo DRA en el proceso de software?



Como se puede observar los encuestados se encuentran divididos, pero a la hora de preguntar en qué consiste este modelo se obtuvieron respuestas como:

“Me imagino que es algo parecido a la metodología de desarrollo eXtremo, donde no existe un proceso de requisitos, sino que se desarrollan directamente las necesidades del usuario final.”

“Modelo rápido de desarrollo de aplicaciones el cual consiste en módulos específicos para cada etapa del desarrollo.”

“Es similar al que llaman modelo de prototipado.”

Aunque fueron menos los que respondieron que sí conocían este modelo, hubo respuestas más acertadas:

“Proceso que busca aumentar la productividad removiendo o minimizando ciertas etapas de procesos de software complejos para reducir tiempos y costos de desarrollo, con una mayor y continua interacción entre clientes y desarrolladores.”

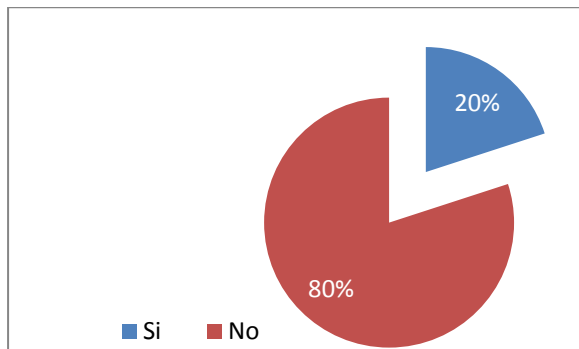
“Es un modelo basado en desarrollos ágiles, el cual permite un ágil desarrollo completamente funcional en cortos periodos de tiempo.”

Se puede concluir que los ingenieros tienen un mayor conocimiento en este modelo y lo manejan, tienen claro el concepto de “aplicación de desarrollo rápido”.

Luego se les preguntó por otro modelo utilizado en la actualidad.

¿Sabe qué es modelo de Cuarta Generación en el proceso de software?

Figura 3. Pregunta 3 ¿Sabe qué es modelo de cuarta generación en proceso de desarrollo de software?



El 80% de los encuestados dice no conocer los modelos de software de Cuarta Generación, lo que lleva a concluir que entre más reciente sea el modelo menos se conoce sobre él; por lo tanto la primera problemática dentro de las empresas es incentivar a sus ingenieros a conocer los avances tecnológicos y las tendencias en el mundo acerca de la Ingeniería de Software y, por otro lado, los ingenieros de software deben saber que el área está en permanente cambio y por tal motivo exige estar constantemente informado y actualizado acerca de los avances en la materia.

Aun así se obtuvieron respuestas muy cercanas a la realidad cuando se les preguntó en qué consiste:

“Herramientas que permiten ahorrar tiempo en el desarrollo.”

“Es una herramienta muy útil para la generación de software, ya que le permite al grupo de trabajo por medio de herramientas especializadas de mapeo hallar dónde se pueden presentar errores, también requiere utilizar herramientas CASE para generador de reportes, base de datos, definidores de pantallas, generadores de código, generador de gráficas, hoja de cálculo.”

“Es el modelo actualmente utilizado por la mayoría de desarrollos, entre los cuales se utilizan métodos como el RUP y el CDM, entre los que tenemos la generación de métodos que permiten implementar la programación orientada a objetos y la documentación estándar, entre la que tenemos el UML.”

“Facilitan al que desarrolla el software la propiedad de especificar algunas características del mismo a alto nivel.”

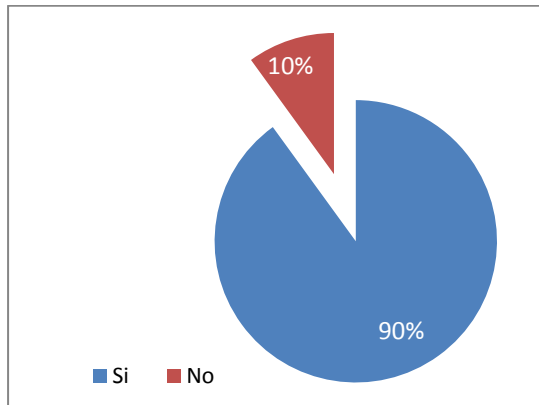
Se puede discernir acerca de estas definiciones dadas pero dado las herramientas actuales que permiten el trabajo de este modelo, lo más aconsejable es utilizarlas para obtener código en alto nivel, ya que aún al día de hoy existen muchos inconvenientes cuando genera código de lógica negocio y es difícil de interpretar y mantener. Por lo tanto muchas de estas respuestas se pueden tomar como acertadas.

2.2 Preguntas referente a la empresa

Una vez diferenciado que la mayoría de ingenieros no tienen totalmente claros los conceptos de los modelos de software, aunque tienen alguna noción de ellos, en las preguntas siguientes se mostrará qué piensan acerca del modelo implementado en el lugar donde laboran:

¿Utilizan modelos de procesos de software en la empresa?

Figura 4. Pregunta 4 ¿ Utilizan modelos de proceso de desarrollo de software en la empresa?.

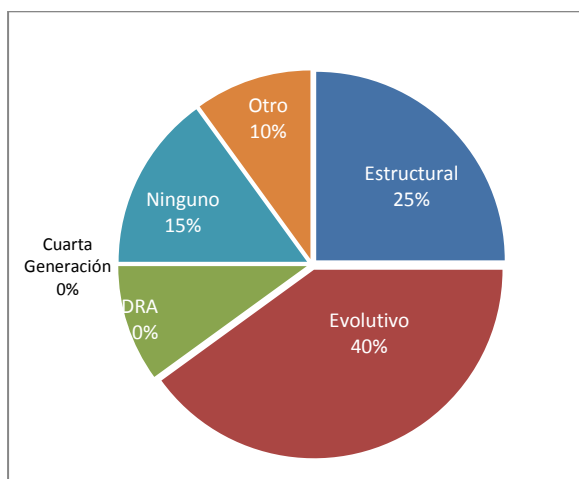


En este momento se evidencia que la mayoría de empresas aplican un modelo de software, lo cual implica que las empresas tienen claro que uno de los factores fundamentales para la realización de productos de calidad debe partir de un modelo de software.

Ahora bien, ¿se podrá afirmar que los ingenieros de las empresas conocen cómo funciona ese modelo? Esta pregunta se responde en los siguientes puntos de la encuesta:

¿Qué modelo utiliza en su empresa?

Figura 5. Pregunta 5 ¿Qué modelo de software emplea en su empresa?



La mayoría mencionó el evolutivo pero lo más interesante es que cinco de ellos mencionaron estructural, el cual es el más antiguo de los modelos.

A la hora de preguntar en qué consiste el modelo de la empresa, de los ingenieros que previamente dijeron estructural, tres de ellos contestaron:

“No sabemos pero tienen las fases de análisis, desarrollo e implementación.”

El cuarto contestó:

“En la empresa se trabajan los desarrollos de manera secuencial, en donde primero se realiza un requerimiento con los interesados, en seguida se realiza el desarrollo, posteriormente se realizan pruebas internas y pruebas de usuario y finalmente es implementado en producción.”

Esta última es la más acertada con un modelo lineal, pero ¿acaso estas fases no hacen parte también del evolutivo?, y si se observa más detenidamente, así como está descrito se podrán aplicar perfectamente a un modelo DRA ya que no se tiene en cuenta el diseño.

El quinto contestó:

“Se aplican las fases del estructural lineal.”

Cuatro de los cinco que respondieron estructural pertenecen a la misma empresa, por lo tanto se puede establecer una veracidad en la información. Lo crítico es que ninguno tiene un conocimiento real del modelo, lo cual es una clara deficiencia.

Ahora se revisarán las respuestas dadas por las personas que escogieron el modelo evolutivo.

Algunas fueron:

“Todo lo van cambiando (casos de uso, desarrollo).”

“Los analistas toman los requerimientos del cliente y se comienza el desarrollo que luego es complementado con nuevos requerimientos o con ajustes de requerimientos sobre lo que ya se ha implementado.”

“Se parte de casos de uso con una especificación básica y se desarrolla, luego se introducen los cambios a medida que el cliente va generando nuevas especificaciones o modificaciones al caso de uso.”

Se observa en las respuestas que algunos ingenieros no tienen ni idea de cómo se implementa el modelo, y los dos que respondieron más estructuralmente se puede decir que sus explicaciones son más dadas a un modelo DRA que evolutivo, ya que son más desarrollos rápidos porque no se tienen la fase de diseño y evaluación de riesgos, simplemente se pasa de los requisitos y requerimientos del cliente al desarrollo directo sobre la aplicación.

Con respecto a las respuestas de los que escogieron DRA, algunas fueron:

“Requerimientos, desarrollo, pruebas e implementación.”

“El analista interactúa con el cliente, luego se sacan los requerimientos y se pasa a desarrollo en forma de historia, luego de realizar el desarrollo pasa a pruebas y por último se implementa.”

¿Estas dos respuestas acaso no se parecen a las descritas para estructural y evolutivo?

Como consecuencia de las primeras preguntas realizadas abarcando los conceptos de los modelos, podría afirmarse que se puede tener la noción del modelo, sin embargo en algunos casos este concepto básico no es suficiente para saber cómo es su aplicación o implementación, desde el punto de vista del proceso y de sus fases.

Ahora bien, de los que contestaron otros, algunos se refirieron a cascada y argumentaron lo siguiente:

“El modelo de cascada define varias etapas desde la planeación, análisis, desarrollo, pruebas e implementación.”

En este aspecto se puede afirmar que:

- Los ingenieros de las empresas encuestadas claramente no tienen en cuenta el diseño como una fase primordial de los modelos de software.
- Las empresas no usan diseño en sus aplicaciones.
- Los ingenieros desconocen cómo se aplican los modelos de software en la vida real, aunque algunos tienen los conceptos claros o nociones de ellos.

- Algunos ingenieros confunden modelos de software con métodos de software y de ahí vienen algunos errores dentro de la aplicación en el proyecto.

Esto implica una problemática a nivel de la rama de la Ingeniería de Software, ya que si los ingenieros de esta área no tienen claro cómo se implementan los distintos modelos, entonces:

¿Cómo pueden discernir entre uno y otro según la problemática que se les presente?

¿Cómo se puede definir un proceso cuando el ingeniero no sabe cómo se implementa?

Roger Pressman define el proceso de la siguiente forma:

“A un nivel detallado, el proceso que adoptemos depende del software que estamos construyendo. Un proceso puede ser apropiado para crear software de un sistema de aviación, mientras que un proceso diferente por completo puede ser adecuado para la creación de un sitio web.”

Y en la encuesta se observa que aunque los ingenieros confunden los diferentes modelos de software algunos tienen el concepto, pero casi todos en forma generalizada.

Además en la implementación, siempre se utiliza el mismo proceso y modelo dentro de la compañía, lo cual es preocupante. Aún más, para las casas de software donde se desarrollan diferentes proyectos es necesario evaluar qué modelo es mejor aplicar.

Para el caso del hospital se puede realizar una excepción, ya que el negocio no cambia y los desarrollos se orientan a mantener el software existente y mejorar algunas funcionalidades de una ERP⁴ para manejo de historias clínicas y módulos administrativos, desarrollada por la Fundación Cardiovascular FCV⁵.

Hubiera sido válido también que respondieran que en la empresa utilizan modelos genéricos. Según Ian Somerville: “Los modelos genéricos no son descripciones definitivas de procesos de software;

⁴ Enterprise Resource Planning (sistemas gerenciales de información).

⁵ Fundación Cardiovascular. Creada en el año 1988: <http://www.fcv.org.site/>

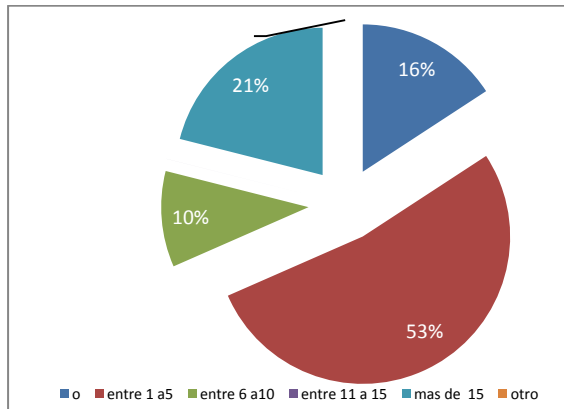
⁶ Hospital San Ignacio. Fundado en el año 1942: <http://www.husi.org.co/>

⁷ Fundación Cardiovascular. Fundada en el año 1988: <http://www.fcv.org.site/>

sin embargo, son abstracciones útiles que pueden ser utilizadas para explicar diferentes enfoques del desarrollo de software.”[5]

¿Cuántos proyectos exitosos se han hecho con el modelo aplicado a la empresa?

Figura 6. Pregunta 6 ¿Cuántos proyectos se ha realizado con el modelo aplicado?



Solo cuatro de ellos respondieron más de 15 (los cuales hacen parte de la misma compañía), lo preocupante es que la mitad están entre 1 a 5 pero a su vez es sorprendente que tengan algún proyecto de software que haya sido exitoso dadas las problemáticas descritas anteriormente.

En la Ingeniería de Software un proyecto es exitoso y tiene calidad en gran parte por la implementación de los modelos de software.

Curiosamente los que respondieron que ningún proyecto ha sido exitoso son los mismos que respondieron que no aplicaban un modelo de software dentro su compañía. Lo que sugiere que el número de proyectos exitosos es una variable que depende directamente de la aplicación de modelos de software.

También es de resaltar que a pesar de que una de las casas de software entrevistadas tiene implementado el modelo de madurez de capacidades integrado CMMI[6] en nivel 5, las respuestas sobre el modelo de software utilizado no tienen exactitud.

III. CONCLUSIONES

Los modelos de software se aplican según el contexto y circunstancias del proyecto, y cada cual tiene ciertas ventajas y desventajas. Depende de los encargados del área de desarrollo decidir el modelo apropiado.

En la actualidad la mayoría de los ingenieros de las empresas no tienen claro los conceptos de los modelos de software existentes y a medida que se indaga sobre los modelos más “recientes” se evidencia un mayor desconocimiento.

Dado lo anterior la consecuencia es que los ingenieros no saben cómo implementar los modelos y desconocen qué modelo se implementa en su empresa siendo un factor crítico.

Queda también pendiente para una investigación la demostración de la hipótesis siguiente: “El número de proyectos exitosos es una variable que depende directamente de la aplicación de modelos de software”.

Por otra parte, las empresas siempre aplican el mismo modelo de software para todos los proyectos sin evaluar cuál es el más adecuado, sin considerar que cada proyecto es diferente y se debe evaluar cuál modelo es más eficiente o acorde al proyecto.

En términos generales, no se tiene un concepto claro acerca de los modelos de software y muchas veces se confunden con los métodos de software, causando inconvenientes críticos, ya que si no se tienen claros los conceptos ¿cómo se puede lograr un buen producto de software?

Ahora bien, queda en el tintero de quién es la culpa del desconocimiento de los ingenieros acerca de la implementación de los modelos de software, puede ser: ¿culpa de la academia por no formar a los futuros ingenieros con base a estos conceptos? o ¿culpa de los ingenieros por no comprometerse con la rama de la ingeniería de software y estar investigando acerca de los nuevos modelos y su implementación? o ¿culpa de las empresas por no capacitar a sus ingenieros y estar a tono con los nuevos modelos de software? Estas preguntas quedan propuestas para futuras investigaciones que indaguen más sobre esta problemática.

REFERENCIAS

- [1] Pressman, R. (2002). *Ingeniería de software un enfoque práctico*. 5ª ed. Madrid: MacGrawHill, cap. 1, 2, 3.
- [2] Schwaber, K. & Beedle, M. (2002). *Agile Software Development with Scrum*. New Jersey: Prentice Hall, cap. 5.
- [3] Ford, G. & Gibbs, N. (1996). *A Mature Profession of Software Engineering* (pp. 25-30). Pittsburgh, PA: SEI, Carnegie Mellon University.

- [4] Quintero, J. B.; Duitama, J. F. (2011) Reflexiones acerca de la adopción de enfoques centrados en modelos en el desarrollo de software. *Ingeniería y Universidad*, 15(1) 219-143.
- [5] Sommerville, I. (2002). *Ingeniería del software*. Madrid: Pearson Educación.
- [6] Oré Bravo, A. (2008). *Introducción al CMMI*. Recuperadode:
http://www.calidadyssoftware.com/-otros/introduccion_cmmi.php

