

## TECNOLOGÍA

## Bits, Ciencia y Sociedad

abr  
27

## Productividad en la Industria TIC

Posteado a las 14:30 | 5 comentarios



Cecilia Bastarrica, profesora del Departamento de Ciencias de la Computación, FCFM, U. de Chile.

En 1986, Frederick Brooks, ganador del Turing Award 1999, escribió un artículo clásico de ingeniería de software titulado: **"No hay balas de plata: Esencia y Accidentes de la Ingeniería de Software"**<sup>[1]</sup>. La idea allí presentada consistía en que los proyectos de desarrollo de software eran como un hombre lobo: algo conocido e inofensivo que podía convertirse de un momento a otro en una bestia que sólo podría ser aniquilada con una bala de plata. Pero el artículo de Brooks postulaba que no existían balas de

plata para el desarrollo de software, ni métodos ni técnicas harían que la productividad fuese creciente y controlable.

Hoy en día, 25 años más tarde, siguen existiendo empresas desarrolladoras de software, y el software invade pervasivamente todos los ámbitos de nuestras vidas. Es de suponer entonces que su desarrollo es un negocio rentable. ¿Habríamos hallado la bala de plata para dominar a la bestia? ¿Cuál es el enfoque seguido entonces para lograr la productividad necesaria para hacer del desarrollo de software una tarea controlable?

Probablemente han tenido más impacto en la productividad del desarrollo de software en las últimas décadas los aspectos metodológicos que los tecnológicos. Si bien hoy existen en el mercado generadores de código o herramientas de apoyo al proceso de pruebas que permiten mejorar la productividad, estos abordan dificultades accidentales del desarrollo de software, en términos de Brooks, y por lo tanto las ganancias que proporcionan son marginales.

En cambio, estandarizar los procesos de desarrollo o definir buenas prácticas y seguirlas sistemáticamente parece haber sido mucho más efectivo. En esta línea han surgido dos tendencias, aparentemente antagónicas pero ambas igualmente atractivas y exitosas, que se basan en la idea de que siguiendo un buen proceso logramos obtener un buen producto, y si bien esto no es una consecuencia lógica, existe una alta correlación.

Por un lado, existe la idea de definir estándares o modelos de madurez que definen las áreas de un proceso de desarrollo que toda empresa debe tener. Estas áreas han sido obtenidas de la experiencia práctica de muchos años. En esta tendencia están los estándares ISO de desarrollo de software y el modelo CMMI.

ISO es un estándar que determina una serie de procesos principales, de soporte y administrativos, que debe abordar una empresa para el desarrollo de software. La versión más reciente es ISO/IEC 12207 del año 2008. Una empresa puede certificarse en este estándar definiendo una forma de abordar cada uno de los procesos establecidos, y aplicándolos dentro de la empresa para el desarrollo de sus proyectos de software.

Otra alternativa dentro de la línea de definición de procesos estandarizados es CMMI. Éste fue originalmente propuesto por el Software Engineering Institute (SEI) de la Universidad de Carnegie Mellon, en 1993, como Capability Maturity Model (CMM), y reformulado a partir de 2006 como Capability Maturity Model Integration (CMMI). La última versión es de noviembre de 2010. Este modelo define una serie de áreas de proceso que las empresas pueden implementar de manera incremental. Las empresas que eligen implementar CMMI no se certifican, sino que se evalúan con un nivel entre 1 y 5 que corresponde al grado de madurez alcanzado: inicial, repetible, definido, gestionado y optimizado.

Tanto ISO como CMMI guían a las empresas en la definición de su propio proceso de desarrollo, proporcionando un compendio de prácticas que han sido efectivas en la práctica. Este es el camino de los procesos estructurados y en general es la estrategia adoptada por empresas grandes que requieren tener procedimientos y documentos estandarizados para que todos sus empleados puedan comunicarse en un lenguaje común. Teniendo establecido cuáles son todos los roles, los documentos, y las actividades que ocurrirán durante el desarrollo se puede hacer planificación y basar una empresa comercial en estos términos. Sin embargo, los detractores de estos métodos esgrimien que es una forma de fosilizar las ineficiencias dentro de las organizaciones, o la única forma en que desarrolladores medianos puedan construir algo útil.

Como alternativa, han surgido hace casi dos décadas los llamados métodos ágiles de desarrollo de software. Entre los más conocidos se encuentran Programación Extrema o Scrum. Estos métodos proponen una forma alternativa cuyo resultado depende más de las personas que de los procesos. Aquí no hay actividades ni documentos predefinidos que deben completarse, sino que es responsabilidad de los desarrolladores decidir qué hacer para resolver cada problema enfrentado. Estos métodos promueven una serie de prácticas efectivas pero que no agregan burocracia al desarrollo, tales como: desarrollo incremental, programación de a pares, desarrollo dirigido por pruebas, o integración del cliente al proceso de desarrollo. Esta estrategia permite optimizar el proceso de desarrollo permitiendo obviar las actividades innecesarias en cada caso. Claramente, la calidad de los resultados depende de la calidad de las decisiones tomadas, y por lo tanto de las competencias de los desarrolladores. La comunicación en estos grupos no está mediada por documentos y por lo tanto no serían métodos apropiados para grandes grupos de desarrollo.

Como parte de la agenda digital de Chile publicada en marzo de 2004 se establecía que para mejorar la industria TIC de Chile era necesario promover la certificación de las empresas desarrolladoras de software. Es decir, se tenía una postura partidaria de las estrategias estandarizadas. Desde entonces se ha difundido entre las empresas la motivación por la definición de los procesos de software aunque no tanto la certificación o evaluación. Sin embargo, y tal como Frederick Brooks ya lo sugería en su artículo original, una de las formas en que se podría promover la productividad es darle un rol preponderante a los super desarrolladores, tal como sucede en las metodologías ágiles. ¿No será entonces más efectivo dedicar los esfuerzos a la educación de muy buenos desarrolladores y promover las metodologías ágiles para lograr un mayor desarrollo de la industria TIC en Chile? La disyuntiva está aún planteada y la respuesta no es clara.

[1] Brooks, Fred P. (1986). "No Silver Bullet—Essence and Accident in Software Engineering". *Proceedings of the IFIP Tenth World Computing Conference*. 1069–1076.

Tags: Bastarrica, Cecilia, Industria TIC, Productividad



BLOGS QUE CITAN ESTE POSTEO

comentarios 5 comentarios

COMENTAR

Visualizando 1-5 de 5

VER TODOS »



## perfil del autor



El blog Bits, Ciencia y Sociedad de la sección de Tecnología de Terra es un espacio donde académicos del Departamento de Ciencias de la Computación de la Universidad de Chile hablarán de la Tecnología y su impacto político y social en nuestro país.

Aquí escribirán semanalmente José Miguel Piquer, Claudio Gutiérrez, Juan Álvarez, Tomás Barros, Nancy Hitschfeld, Benjamin Bustos, Alejandro Hevia, Pablo Barceló y Cecilia Bastarrica.

## posteos

VER: [MÁS RECIENTES](#) [MÁS COMENTADOS](#)

## Productividad en la Industria TIC

27 de Abril de 2011 - 14:30

## Información geográfica para todos: un derecho básico

19 de Abril de 2011 - 15:26

## ¿El lenguaje importa?

13 de Abril de 2011 - 10:25

## El acceso a Internet en Chile

5 de Abril de 2011 - 11:19

## Transferencia tecnológica: buscando fórmulas

22 de Marzo de 2011 - 11:50

BUSCAR



## últimos comentarios

“ Hay algo que pienso que esta más o menos... ”

Juan Pablo Lizana en Productividad en la Industria TIC >

“ Muy buen artículo profesora y excelente idea la de... ”

Alfredo Diaz Puentes en Productividad en la Industria TIC >

“ Excelente resumen Cecilia! Me llama mucho la atención los dos... ”

José Miguel Vives en Productividad en la Industria TIC >

“ Tal como el nombre CMMI, este es un proceso... ”