

MODELO DE MADUREZ DE LA CAPACIDAD DEL SOFTWARE

Guillermo Choque Aspiazu
gchoquea@gmail.com

Resumen

Para llevar los proyectos informáticos a buen término no basta con aplicar ingeniería del software. Hay que hacerlo racionalmente, sin despilfarrar recursos y utilizando en cada momento las técnicas más adecuadas. El Modelo de Madurez de la Capacidad del Software desarrollado por el Instituto de Ingeniería del Software, en la prestigiosa Universidad Carnegie Mellon de los Estados Unidos es la referencia más solvente para ponderar el grado de adecuación del proceso del software en una organización cuya meta normal, en estos tiempos, es contar con un producto software de calidad.

Palabras clave: Modelo de madurez, inicial, repetible, definido, gestionado, optimizado.

1. INTRODUCCION

Desde el momento en que una organización expresa una necesidad en materia de tecnologías de la información a un fabricante del software se inicia un proceso complejo que involucra a un conjunto de personas con funciones heterogéneas que deben coordinarse para satisfacer esa necesidad. La experiencia menciona que cuanto mayor es la envergadura de las tareas a desarrollar más incertidumbre se genera en aspectos clave como: costos, la fecha de entrega y la calidad del resultado. En gran medida, esta incertidumbre surge de la no aplicación de técnicas de ingeniería del software. No es infrecuente tener noticias de grandes proyectos que fracasan por haberse acometido sin rigor metodológico: los requisitos no han sido especificados correctamente, no se dispone información veraz del estado del proyecto porque no se mide su progreso, la documentación es ambigua o insuficiente, la comunicación entre los miembros del equipo no es la mejor posible, no se ejecutan tareas de reducción, supervisión y gestión del riesgo, se presta poca atención a las estrategias de prueba, las fechas de entrega las fija el cliente como requisito inicial en lugar de permitir al proveedor usar técnicas formales de estimación y planificación, llega el momento en que todo el equipo del proyecto se ve abocado a trabajar incontables horas a destajo para subsanar "imprevistos", las prisas llevan a abandonar la planificación y a obsecarse en terminar como sea [López, 2004].

Existe cierta tendencia a evitar la ingeniería del software por considerar que burocratiza en exceso

el desarrollo de proyectos, quizá por desconocimiento de algunas de las técnicas o por la sensación de pérdida de tiempo que genera la realización de tareas consideradas accesorias que no impliquen la mera construcción del producto final. Pero la realidad establece que cuando se acometen grandes proyectos sin rigor metodológico se desemboca normalmente en el más estrepitoso de los fracasos, sobre todo en proyectos grandes: los costos se disparan, las fechas no se cumplen y la calidad del producto no es la esperada. La correcta utilización de la ingeniería del software, aunque en un primer momento pueda parecer tediosa, conduce a un abaratamiento de los costos y al cumplimiento de los plazos de entrega del producto software.

Sin embargo, para llevar los proyectos a buen término no basta con aplicar ingeniería del software. Hay que hacerlo racionalmente, sin despilfarrar recursos y utilizando en cada momento las técnicas más adecuadas. El Modelo de Madurez de la Capacidad del Software desarrollado por el Instituto de Ingeniería del Software de los Estados Unidos es la referencia más solvente para ponderar el grado de adecuación del proceso del software en una organización. En este artículo se ofrece las líneas básicas para transmitir una visión global del mismo. Al final del artículo se proporcionan las referencias bibliográficas para profundizar en los contenidos.

2. PROCESO DEL SOFTWARE

Un proceso del software se puede caracterizar de acuerdo a lo que se muestra en la figura 1.

Normalmente se establece un marco común del proceso definiendo un número pequeño de actividades del marco de trabajo que son aplicables a todos los proyectos del software, con independencia de su tamaño o complejidad. Un número de conjuntos de tareas¹ que permiten que las actividades del marco de trabajo se adapten a las características del proyecto de software y a los requisitos del equipo del proyecto. Finalmente, las actividades de protección² abarcan el modelo de procesos. Las actividades de protección son independientes de cualquier actividad del marco de trabajo y aparecen durante todo el proceso [Pressman, 2002].

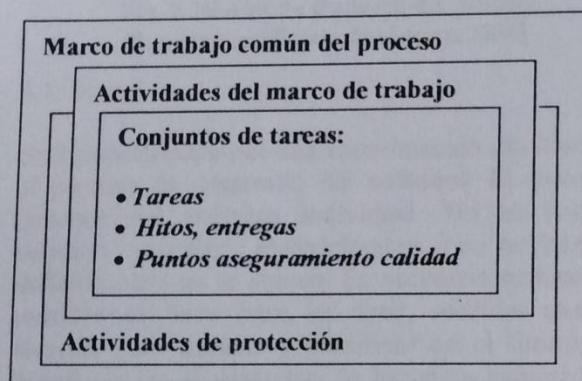


Fig. 1. Proceso del software
Fuente: [Pressman, 2002]

3. MODELO DE CAPACIDAD Y MADUREZ DEL SOFTWARE

En el Modelo de Madurez de la Capacidad del Software³ del SEI se define un conjunto de *áreas clave del proceso*, que describen las funciones de ingeniería del software que deben llevarse a cabo para el desarrollo de una buena práctica, agrupadas en cinco niveles inclusivos. Estos niveles sirven de referencia para el conocimiento del estado de la madurez del proceso del software en la organización. Mediante un amplio conjunto de métricas se determina la calidad de cada una de las áreas clave, obteniéndose una visión precisa del rigor, la eficacia y la eficiencia de la metodología de desarrollo de una organización productora de software. Cada una de las áreas

¹ Cada uno es una colección de tareas de trabajo de ingeniería del software, hitos de proyectos, productos de trabajo y puntos de garantía de calidad.

² Tales como garantía de calidad del software, gestión de configuración del software y medición.

³ Software Capability Maturity Model, SW-CMM.

está organizada en cinco secciones, denominadas *características comunes*:

- Compromiso de realización.
- Capacidad para llevarla a cabo.
- Actividades que hay que realizar.
- Medición y análisis.
- Verificación de la implementación.

En cada característica común se especifican unas *prácticas clave*, que son normas, procedimientos y actividades cuya realización lleva a la consecución de los objetivos del área. En algunos casos se detallan subprácticas más específicas, guías e interpretaciones de la práctica y, cuando procede, ejemplos y referencias cruzadas a otras prácticas. Por ejemplo, las prácticas de la característica *medición y análisis* describen las medidas que se han de realizar sobre el área de proceso correspondiente. Asimismo, el SEI define *indicadores clave*, que son aquellas prácticas clave o componentes de prácticas clave que ofrecen una visión mayor de la consecución de los objetivos de un área clave de proceso. Tal como hemos dicho al principio de este apartado, los niveles en los que se agrupan las áreas claves de proceso son inclusivos: para alcanzar uno es necesario haber alcanzado (y mantener) todos los anteriores

4. NIVELES DE MADUREZ DEL PROCESO

En los últimos años, se ha depositado mucho énfasis en la "madurez del proceso". El Instituto de Ingeniería del Software⁴ (SEI) ha desarrollado un modelo completo que se basa en un conjunto de funciones de ingeniería del software que deberían estar presentes conforme las organizaciones alcanzan diferentes niveles de madurez del proceso. Para determinar el estado actual de madurez del proceso de una organización, el SEI utiliza un cuestionario de evaluación y un esquema de cinco grados. El esquema de grados, que se observa en la figura 2, determina la capacidad de madurez [Paulk et al., 1993] que define las actividades clave requeridas en los diferentes niveles de madurez del proceso. El enfoque del SEI proporciona una medida de la efectividad global de las prácticas de la ingeniería del software de una compañía y establece cinco niveles de madurez del proceso que son los

⁴ Software Engineering Institute.

siguientes: inicial, repetible, definido, gestionado y optimización [Pressman, 2002].

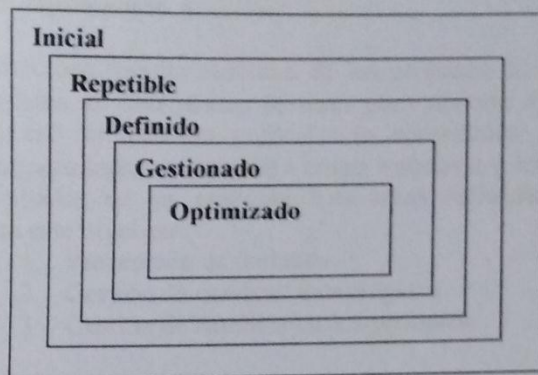


Fig. 2. Niveles de madurez del proceso
Fuente: modificado de [Lopez, 2004]

4.1. Inicial

Está caracterizado por una aproximación intuitiva al proceso de desarrollo del software. El éxito depende del esfuerzo individual. No se han definido procesos metodológicos, o se han definido pero no se siguen. Es necesario realizar *medidas de línea base*, es decir, medidas que servirán para estimar y planificar en el futuro. Asimismo, es el momento de hacer un esfuerzo de estructuración y control en el proceso.

4.2. Repetible

La madurez metodológica de la organización permite estimar fiablemente el tamaño funcional o físico del sistema, así como recursos, esfuerzo, costes y calendario. Se han sentado las bases para repetir éxitos anteriores en proyectos con aplicaciones similares. Las áreas clave de proceso definidas en este nivel, cuyo estado se puede conocer mediante diversas métricas, son las siguientes:

1. Gestión de requisitos.
2. Planificación del proyecto software.
3. Seguimiento y control del proyecto.
4. Gestión de la subcontratación del software.
5. Aseguramiento de la calidad del software.
6. Gestión de la configuración del software.

Por ejemplo, en el área 6 se pueden medir el número de peticiones de cambio procesadas por unidad de tiempo y los fondos empleados en gestión de configuración.

4.3. Definido

Se conoce la forma de construcción del sistema. El proceso del software de las actividades de gestión e ingeniería se documenta y se estandariza. Las actividades intermedias están bien definidas, y por tanto se pueden examinar y medir. Por ejemplo, se pueden medir la complejidad ciclomática del código, los defectos descubiertos o la densidad de errores por producto. Además es posible detectar tempranamente posibles problemas y aplicar una adecuada gestión de riesgo.

Las áreas clave definidas en este nivel son:

1. Desarrollo y mejora de los procesos de la organización.
2. Definición de los procesos de la organización.
3. Programa de formación.
4. Gestión integrada del software.
5. Ingeniería de producto software.
6. Coordinación intergrupos.
7. Revisión conjunta.

Por ejemplo, en el área 1 se podría medir el esfuerzo empleado en las actividades de evaluación, desarrollo y mejora de los procesos de la organización comparado con el plan. En el área 2 se podría medir el coste de las actividades de definición del proceso.

4.4. Gestionado

Se añade la gestión a un proceso definido. Se usa realimentación desde las primeras actividades del proyecto para seleccionar prioridades en las actividades actuales y conocer cómo se emplean los recursos. Los efectos de los cambios en una actividad se pueden seguir en otras. Se recopilan medidas detalladas del proceso del software y de la calidad del producto. En definitiva, se evalúa la efectividad de las actividades del proceso. Por ejemplo, se podría medir cuánto se está produciendo para ser reutilizado, cuánto se está reutilizando de proyectos anteriores, cómo y cuándo son descubiertos los defectos y la relación entre fechas de finalización de los módulos y fechas previstas. Las áreas clave definidas en este nivel son dos:

1. Gestión cuantitativa del proyecto.

2. Gestión de calidad del software.

4.5. Optimizado

Existe una mejora continua de los procesos. Las medidas de actividades se usan para mejorar el proceso, eliminando y añadiendo actividades y reorganizando su estructura como respuesta a los resultados de las medidas. Las áreas definidas para este nivel son:

1. Prevención de defectos.
2. Gestión de cambios tecnológicos
3. Gestión de cambios en los procesos.

Por ejemplo, en el área 2 se podrían medir los efectos de la implementación de los cambios tecnológicos comparados con los objetivos. En el área 3 se podría medir el número de propuestas de mejora enviadas por departamento. Además del Modelo de Madurez de la Capacidad del Software existen el Modelo de Madurez de la Capacidad en la Adquisición de Software⁵ y el Modelo de Madurez de la Capacidad de las Personas⁶ [López, 2004].

5. AREAS CLAVE DEL PROCESO

Los cinco niveles definidos por el SEI se obtienen como consecuencia de evaluar las respuestas del cuestionario de evaluación basado en el Modelo de Capacidad y de Madurez. Los resultados del cuestionario se refinan en un único grado numérico que proporciona una indicación de la madurez de proceso de una organización.

El SEI ha asociado áreas claves del proceso (ACPs) a cada uno de los niveles de madurez. Las ACPs describen esas funciones de la ingeniería del software (por ejemplo: planificación del proyecto de software, gestión de requisitos) que se deben presentar para satisfacer una buena práctica a un nivel en particular. Según Pressman (2002), cada ACP se describe identificando las características siguientes:

- Objetivos: los objetivos globales que debe alcanzar la ACP
- Compromisos: requisitos (impuestos en la organización) que se deben cumplir

para lograr los objetivos y que proporcionan una prueba del intento por ajustarse a los objetivos.

- Capacidades: aquellos elementos que deben encontrarse (organizacional y técnicamente) para permitir que la organización cumpla los objetivos.
- Actividades: las tareas específicas que se requieren para lograr la función ACP.
- Métodos para supervisar la implementación: la manera en que las actividades son supervisadas conforme se aplican.
- Métodos para verificar la implementación: la forma en que se puede verificar la práctica adecuada para la ACP.

Se definen dieciocho ACPs (descritas mediante la estructura destacada anteriormente) en el modelo de madurez y se distribuyen en niveles diferentes de madurez del proceso. Las ACPs se deberían lograr en cada nivel de madurez del proceso⁷.

Nivel 2 de Madurez del Proceso

- Gestión de configuración del software
- Garantía de calidad del software
- Gestión de subcontratación del software
- Seguimiento y supervisión del proyecto del software
- Planificación del proyecto del software
- Gestión de requisitos

Nivel 3 de Madurez del proceso

- Revisiones periódicas
- Coordinación entre grupos
- Ingeniería de productos del software
- Gestión de integración del software
- Programa de Formación
- Definición del proceso de la organización
- Enfoque del proceso de la organización

Nivel 4 de Madurez del Proceso

- Gestión de calidad del software
- Gestión cuantitativa del proceso

Nivel 5 de Madurez del Proceso

⁵ Software Acquisition Capability Maturity Model (SA-CMM).

⁶ People Capability Maturity Model (P-CMM).

⁷ Se debe tener en cuenta que las ACPs son acumulativas. Por ejemplo, el nivel 3 de madurez del proceso contiene todas las ACPs del nivel 2 más las destacadas para el nivel 1.

- Gestión de cambios del proceso
- Gestión de cambios de tecnología
- Prevención de defectos

Cada una de las ACPs se definen con un conjunto de prácticas clave que contribuyen a cumplir estos objetivos. Las prácticas clave son normas, procedimientos y actividades que deben ocurrir antes de que se haya instituido completamente un área clave de proceso. El SEI define a los indicadores clave como "aquellas prácticas clave o componentes de prácticas clave que ofrecen una visión mejor para lograr los objetivos de un área clave de proceso". Las cuestiones de valoración se diseñan para averiguar la existencia (o falta) de un indicador clave [Pressman, 2002]

6. MODELO DE CAPACIDAD Y MADUREZ INTEGRAL

El modelo de capacidad y madurez integral⁸ es un conjunto de modelos elaborados por el SEI que permiten obtener un diagnóstico preciso de la madurez de los procesos relacionados con las tecnologías de la información de una organización, y describen las tareas que se tienen que llevar a cabo para mejorar esos procesos.

Los módulos CMMI son extractos de los modelos CMMI a los que se han añadido posibles pruebas a realizar, y sirven de base para emprender la mejora de procesos. Existen actualmente cuatro modelos CMMI, que contemplan los procesos de mejora en las diversas áreas de los sistemas de información, de manera que la organización deberá elegir el que más se ajuste a sus necesidades:

- CMMI-SE/SW/IPPD/SS
- CMMI-SE/SW/IPPD
- CMMI-SE/SW
- CMMI-SW

Para cada modelo hay una representación *continua* y otra *por etapas*. Las diferencias pueden ser observadas en la tabla 1.

7. CONCLUSIONES

En esta propuesta se comenta que para llevar los proyectos informáticos a buen término no basta con aplicar el proceso de la ingeniería del software, se debe aplicar un proceso racional, sin despilfarrar recursos y utilizando en cada

momento las técnicas más adecuadas. El CMM desarrollado por el Instituto de Ingeniería del Software, en la prestigiosa Universidad Carnegie Mellon de los Estados Unidos constituye la referencia más solvente para ponderar el grado de adecuación del proceso del software en una organización cuya meta normal, es contar con un producto software de calidad. Para sustentar esta proposición se muestra la denominada base estructural que sustenta a la tecnología estratificada conocido como proceso. Posteriormente se define el modelo CMM y los niveles de madurez del proceso, luego las áreas claves del proceso y finalmente la evolución del CMM en un modelo de capacidad y madurez integral.

⁸ Capability Maturity Model Integration (CMMI)

Tabla 1: Diferencias entre representación continua y por etapas
Fuente: elaborado según [López, 2004]

REPRESENTACION CONTINUA	REPRESENTACION POR ETAPAS
Las áreas de proceso se organizan por categorías de áreas de proceso.	Las áreas de proceso se organizan por niveles de madurez.
La mejora se mide en niveles de capacidad que reflejan la implantación incremental de un área de proceso particular.	La mejora se mide utilizando niveles de madurez que reflejan la implementación concurrente de múltiples áreas de proceso.
Hay seis niveles de capacidad (0-6).	Hay cinco niveles de madurez (1-5).
Hay dos tipos de prácticas: básicas y avanzadas.	Hay sólo un tipo de prácticas. El concepto de <i>práctica avanzada</i> se consigue por otros medios.
Los niveles de capacidad se usan para organizar las prácticas genéricas.	Las prácticas genéricas se usan según características comunes.
Todas las prácticas genéricas se usan en todas las áreas de proceso.	Sólo se usan en un área de proceso las prácticas aplicables al nivel de madurez.
Existen prácticas genéricas para los niveles de capacidad del 1 al 5.	Existen prácticas genéricas para los niveles de madurez del 2 al 5. Algunas de las prácticas utilizadas en la representación continua se aplican en algunas áreas de proceso.
Existe la posibilidad de obtener el nivel de madurez equivalente al perfil obtenido.	No es posible determinar con qué perfil de la representación continua se corresponde un determinado nivel.

REFERENCIAS BIBLIOGRAFICAS

- Creative Commons, Metodologías de desarrollo del software, España, octubre 2005. Disponible en: <http://www.um.es/docencia/barzana/IAGP/Iagp2.html>
- Gracia Joaquín, CMM-CMMI Nivel 2, Ingenieros del software, Agosto 2005. Disponible en: <http://www.ingenierossoftware.com/calidad/cmm-cmmi.php>
- Intel Information Technology, People Capacity Maturity Model, February 2003. Disponible en: http://cache-www.intel.com/cd/00/00/10/29/102960_wp031201_sum.pdf
- López C., Modelo de madurez de la capacidad del software. Informas No 1. *Revista del Ilustre Colegio de Ingenieros Informáticos de la Región de Murcia*. España. 2004.
- Paulk M.C, G.M. García, M.B. Crisis y M. Bush. *Capability Maturity Model for Software, version 1.1*. CMU/SEI-93-TR-24, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, febrero 1993.
- Pressman, R.S. Ingeniería del Software: Un enfoque práctico. Quinta edición, McGraw-Hill, 2002. Adaptado por Darell Ince.
- Sitio oficial del SEI. <http://www.sei.cmu.edu/>