

Es evidente que se trata de un área altamente compleja. Estos sistemas se basan en servicios que probablemente ya se consideraron complejos cuando se construyeron. Desde nuestro punto de vista esta complejidad es conatural a ellos y la simplificación de una parte solo conlleva un incremento de la complejidad de otra. Esto no está en contra de que la mejor tecnología es invisible y esa complejidad no puede trasladarse al usuario.

Resulta pues imprescindible introducir una nueva disciplina que contemple como misión principal las nuevas necesidades de este nuevo paradigma: el *Urbanismo Software*.

Desde nuestra perspectiva son elementos fundamentales la capacidad de evolución y autonomía de los servicios más allá de la aplicación para la que fueron concebidos, la potenciación de la reutilización, la necesidad de certificar propiedades de calidad de los servicios, incluyendo aspectos no funcionales y avanzar en el modelo de negocio de los servicios software.

Otra necesidad fundamental es la de aportar sólidos fundamentos semánticos en todas las capas, pero especialmente en la especificación de los servicios. Es esencial para las actividades de certificación, interoperabilidad semántica, cartografías de servicios, etc. Por ello preferimos aquellas aproximaciones a partir del proceso de construcción aseguren las propiedades de los servicios a componer sean fiables y predecibles.

No queremos terminar sin indicar que es imprescindible comprender la omnipresencia del software mejor de lo que lo hemos hecho hasta ahora. Al igual que ha habido muchos esfuerzos en comprender los sistemas naturales y los artefactos de otras ingenierías, se necesita desarrollar teorías que permitan conocer y razonar sobre el software, tan complejo o más como los anteriormente mencionados. Solo así se podrá reconciliar las demandas de software en el mercado con su construcción rigurosa.

## Referencias

- [1] R. Allen, R. Douence, D. Garlan: Specifying and Analyzing Dynamic Software Architectures, *Fundamental Approaches to SW Engineering*, 1998
- [2] Bass L.; Clements P.; Kazman R. *SW Architecture in Practice*, 2<sup>nd</sup> Edition, Addison-Wesley, 2003.
- [3] Desreumaux, M.; Oudhri, R.: Information and Software Systems: from Architecture to Urbanism, WISCAI Conference, IFIP, Chapman & Hall 1998

[4] European Science Foundation Forward-look meeting on Nanosciences and the Future of Information Technology, Paris 4-7 April 2005 (report to be published), [www.esf.org/asi](http://www.esf.org/asi)

[5] R. Filman, T. Elrad, S. Clarke, M. Aksit: Aspect-Oriented SW Development, Addison Wesley, 2004

[6] S. Guadarrama, S. Muñoz, C. Vaucheret: Fuzzy Prolog: A new approach using soft constraints propagation, Fuzzy Sets and Systems, FSS 2002

[7] A. Herranz, J.J. Moreno-Navarro: Formal Extreme (and Extremely Formal) Programming, 4th Int. Conf. on Extreme Programming and Agile Processes in Software Engineering, XP 2003, Springer Verlag LNCS 2675, 2003

[8] A. Herranz, J.J. Moreno-Navarro: Rapid Prototyping and Incremental Evolution Using SLAM, 14th IEEE International Workshop on Rapid System Prototyping, RSP 2003

[9] J. Kramer, J. Magee: Self organising software architectures, Proc. 2nd Int. software architecture workshop (ISAW-2) and Int. workshop on multiple perspectives in software development (Viewpoints'96), pp. 35-38, ACM-Press, 1996.

[10] L. Lamport, J.E. Johnson, D.E. Langworthy, F.H. Vogt: Formal Specification of a Web Services Protocol, 1st Int. Workshop on Web Services and Formal Methods (WS-FM 2004), Pisa, 2004

[11] Lehman, M.M., Ramil J.F.: Software Evolution and Software Evolution Processes, *Annals of Software Engineering*, special issue on Software Process-based SW Eng., vol.14, 2002, pp.275 - 309

[12] J. Luque Valdivia: Constructores de la ciudad contemporánea, Ed. Cic Dossat, Madrid, 2004

[13] N.Maya-Fernández, J.J. Moreno-Navarro: Towards incorporating (formal) semantics into Web Services, JSWEB, Granada 2005

[14] B.Meyer: The Grand Challenge of Trusted Components, *Proc. of the 25th Int. Conf. on SW Engineering*, 2003, IEEE Computer Society

[15] P. Oreizy, N. Medvidovic, R.N. Taylor, Architecture-Based Runtime Software Evolution, Int. Conf. on Software Engineering- ICSE'98, 1998.

[16] A. Rashid, P. Sawyer, A. Moreira, J. Araújo. Early Aspects: a Model for Aspect-Oriented Requirements Engineering, Int. Conf. Requirements Engineering (RE 2002), Univ. of Essen, 2002.

[17] M. Shaw, D. Garlan: Software Architecture Perspectives on an Emerging Discipline. *Prentice Hall*, 1996.

[18] S.C.E. Trillas, A. Pradera: A mathematical model for fuzzy connectives and its applications to operational behavioural study, V 516, *Kluwer Int. Series in Engineering and Computer Sciences*, 1999, Ch 4, pp. 307-318.

[19] L. Zadeh: Fuzzy sets as a basis for a theory of possibility, *Fuzzy Sets & Systems* 1(1) 1978,85-121

# Adaptación de las normas ISO/IEC 12207:2002 e ISO/IEC 15504:2003 para la evaluación de la madurez de procesos software en países en desarrollo

Félix García, Francisco Ruiz, Mario Piattini

Grupo Alarcos  
Escuela Superior de Informática  
Universidad Castilla-La Mancha (España)  
(Felix.Garcia, Francisco.RuizG, Mario.Piattini)@uclm.es

Francisco J. Pino  
Grupo I+D en Tecnologías de la Información  
Departamento de Sistemas  
Universidad del Cauca (Colombia)  
fpino@unicauca.edu.co

## Resumen

Para motivar a las empresas del sector informático colombiano a mejorar sus procesos de desarrollo de software, con el objetivo de lograr un nivel de madurez en sus procesos que garantice su competitividad internacional, es necesario adecuar a sus propias características los modelos internacionales reconocidos de mejoramiento, evaluación y calidad. Estos modelos difícilmente pueden ser aplicados a empresas pequeñas debido a su gran inversión en dinero, tiempo y recursos, además de la complejidad de las recomendaciones y un retorno de la inversión a largo plazo. El objetivo de este trabajo es presentar, a Light MECPDS, un modelo ligero de evaluación de la calidad de procesos de desarrollo de software basado las normas ISO/IEC 12207:2002 e ISO/IEC 15504:2003 aplicable a las micro, pequeñas y medianas empresas, de manera fácil y económica, con pocos recursos y en poco tiempo.

**Palabras clave:** Modelos de Evaluación, Framework de Medida, Capacidad del Proceso, Cumplimiento del Proceso, Modelo de Proceso de Referencia, ISO 12207:2002 e ISO 15504:2003.

## 1. Introducción

La industria de software representa una actividad económica de suma importancia para todos los países del mundo, entre ellos Colombia. Ofrece múltiples fuentes de negocio y se perfila como la oportunidad más grande de los países en vía de desarrollo. Pero, en los países latinoamericanos la industria de software es incipiente e inmadura [1], lo cual conlleva a falta de competitividad que a su vez dificulta su crecimiento.

En Colombia las empresas de desarrollo de software no están preparadas para ser

competitivas internacionalmente. El sector informático se enfrenta a una serie de problemas como la dependencia tecnológica del país, el desconocimiento de la importancia que tiene el proceso de desarrollo sobre la calidad del producto y la construcción de software de forma artesanal, empírica y caótica.

A raíz de esto el software desarrollado es de baja calidad, el tiempo de desarrollo es inapropiado, los costos no son competitivos, las actividades de operación y mantenimiento del software son difíciles y hay incremento de la insatisfacción de los clientes y usuarios finales.

Aún con la desventaja competitiva que tiene la industria del software de Colombia, ésta aumenta progresivamente. Se hace necesario generar estrategias para encaminar a Colombia hacia la dirección de los países con gran desarrollo en la industria informática. Una estrategia primordial es desarrollar productos de calidad. La calidad de los productos está íntimamente ligada con la calidad de los procesos utilizados para desarrollarlos. Entonces se hace evidente que para incrementar la calidad del producto las empresas de desarrollo de software del país deben implementar proyectos para la mejora del proceso software.

Asegurar la calidad a través del mejoramiento de los procesos software es un paso que las empresas del país deben dar como respuesta a dos situaciones: la primera por imagen, para poder exportar software e incursionar y mantenerse en un mercado global; la segunda por necesidad, para poder hacer de sus proyectos unidades administrativas eficaces y eficientes.

Una de las características principales de la industria de software de Colombia, es estar compuesta por micro, pequeñas y medianas empresas - PyMES. Estas empresas de software pequeñas tienen serios problemas de madurez en

sus procesos de desarrollo, en muchos casos no existe un proceso real conduciendo a modelos caóticos de operación que afectan toda la empresa [2]. Además estas empresas también planean asegurar la calidad de sus productos a través de la mejora del proceso acreditándose en modelos de calidad del SEI ó ISO [3]. Pero la preparación previa a la acreditación es larga y costosa. Los modelos de mejoramiento, proceso y evaluación de organizaciones como el SEI e ISO están estructurados para ser aplicables a empresas grandes. Difícilmente pueden ser aplicados a empresas pequeñas debido a que un proyecto de mejora supone gran inversión en dinero, tiempo y recursos, además de la alta complejidad de las recomendaciones y que el retorno de la inversión se produce a largo plazo [4][5][6][7].

Es por esto que el proyecto Sistema Integral para el Mejoramiento de los Procesos de Desarrollo de Software en Colombia, SIMEP-SW<sup>1</sup> busca proporcionar a las empresas del sector informático de Colombia las herramientas necesarias para motivarlas a mejorar sus procesos de desarrollo de software con el objetivo de facilitar el posicionamiento y la competitividad en mercados nacionales e internacionales.

El proyecto pretende crear, aplicar y probar un sistema de mejoramiento que integre elementos de modelos de calidad, mejoramiento y evaluación reconocidos internacionalmente, adaptados a las características propias de la industria del software colombiana y que pueda ser replicado a industrias de características similares a nivel nacional e internacional [8]. Se pretende que los proyectos de mejoramiento que realicen las empresas de Colombia sigan un modelo nacional coherente a las características propias de la idiosincrasia y aterrizadas al contexto socio-económico del país.

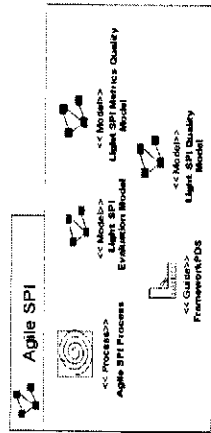


Figura 1. Arquitectura de Agile SPI

<sup>1</sup> El proyecto SIMEP-SW es financiado por Colciencias, Universidad del Cauca y Sitis Ltda.

que en su gran mayoría son PyMES, un modelo basado en las mejores prácticas internacionales fácil de entender, fácil de aplicar y no costoso en su adopción. Pretende apoyar a las organizaciones en la estandarización de sus prácticas, en la evaluación de su efectividad y en la integración de la mejora continua; elevando la capacidad para ofrecer servicios con calidad y alcanzar niveles internacionales de competitividad.

MoProSoft define tres categorías de procesos: Alta Dirección-DIR, Gestión-GES y Operación-OPE. Para cada uno de los procesos específica tres partes: definición general del proceso, prácticas y guía de ajuste. Basa su estrategia de mejora en que la organización debe establecer la estrategia (la cual no es explícita en el modelo) de implantación de los procesos definidos por el modelo. Los procesos deben evolucionar en base a las sugerencias de mejora alcanzando los objetivos del plan estratégico de la organización con metas cada vez más ambiciosas. De esta manera la organización puede ir logrando la madurez a través de la mejora continua de sus procesos.

En el caso de Brasil, se está desarrollando el proyecto "mps Br" [13]. Basado en ISO/IEC 12207:2002, CMMI e ISO/IEC 15504:2003, tiene como objetivo principal definir e implementar un modelo para la mejora de procesos de software. Pretende dar respuesta a la pregunta ¿Cómo mejorar radicalmente los procesos de software en Brasil, con foco en un número significativo de PyMES de forma que estas obtengan un nivel de madurez 2 o 3 a un costo accesible? El proyecto "mps Br", desarrolló dos modelos: un Modelo de Referencia para la mejora del proceso del software – "MR mps" y un Modelo de Negocio para la mejora del proceso del software – "MN mps".

"MN mps" define los elementos e interacciones involucrados para la certificación de la empresa a través de la implementación de "MR mps" de dos maneras: personalizada para una empresa o conjunta entre un grupo de empresas (logrando así costos más accesibles para PyMES).

"MR mps" comprende niveles de madurez y un método de evaluación. Los niveles de madurez están organizados en dos dimensiones: de capacidad y de proceso. La madurez del proceso define 7 niveles. A cada nivel de madurez se le atribuyen áreas de proceso con base en los niveles de CMMI, para posibilitar la implementación gradual y adecuada en las PyMES brasileñas. El

método de evaluación a partir de indicadores asigna un nivel de implementación de una práctica relacionada a un área de proceso.

En los modelos anteriores no se explicita ninguna estrategia de mejora guiada por un proceso de mejora. SIMEP-SW basa su estrategia de mejora en proporcionar a la organización un proceso ágil que guía a un programa de mejora de modelo de evaluación ligero ya que, para poder promover la mejora de los procesos software, es muy importante establecer previamente un marco de evaluación con el fin de conocer sus puntos fuertes y débiles. La evaluación de los procesos software tiene como objetivo detectar aspectos de un proceso software que se pueden mejorar [14].

En cuanto al modelo de evaluación MoProSoft no tiene uno explícito. "MR mps" lo define de la intersección de las dimensiones de madurez y cumplimiento del proceso relacionándolo con el esquema de niveles de madurez de CMMI, en su representación escalonada.

Light MECPDS se basa en ISO 15504:2003, y define un marco de trabajo de medición para dar soporte a la evaluación en las dimensiones de capacidad del proceso y del cumplimiento del proceso. En la dimensión de la capacidad, sólo existen tres niveles de madurez, con el fin de aligerar el modelo y de que pueda ser aplicado a las PyMES. Además sugiere como modelo de proceso de referencia a ISO 12207:2002.

En [15] se presenta un modelo de evaluación de procesos de software basado en ISO/IEC TR 15504-5:1998, orientado a PyMES de desarrollo de software que permite deducir la capacidad de los procesos del ciclo de vida del software. Light MECPDS, utiliza para su definición las normas ISO/IEC 15504:2003 e ISO/IEC 12207:2002 que son las más reciente para procesos de evaluación.

### 3. Normas utilizadas en Light MECPDS

#### 3.1. ISO/IEC 12207:2002

Esta norma presenta un modelo de procesos de referencia del ciclo de vida del software que son fundamentales para una buena ingeniería de software y cubre las mejores prácticas. Los procesos son descritos en términos de lograr los propósitos y resultados. Además precisa las

actividades y tareas requeridas para implementar a alto nivel los procesos para alcanzar las capacidades deseadas para los adquirientes, proveedores, desarrolladores, responsables de mantenimiento y operadores del sistema que contiene el software. El modelo de referencia es también usado para proveer una base común para diferentes modelos y métodos asegurando que la evaluación sea realizada en un contexto común.

3.2. ISO/IEC 15504:2003

Esta norma, denominada "tecnologías de información: proceso de evaluación", está constituida por cinco partes.

La parte 2, guía la evaluación del proceso y la aplicación del proceso de evaluación para el mejoramiento y determinación de la capacidad. Precisa los requisitos mínimos para realizar una evaluación que asegure un nivel de consistencia y capacidad de repetición, y que los resultados de la evaluación sean objetivos, imparciales, repetibles, consistentes y representativos. Identifica el framework de medida para la capacidad del proceso y los requisitos para: el modelo de procesos de referencia, el modelo de evaluación de procesos y la verificación de la conformidad del proceso de evaluación.

El modelo del proceso de evaluación contiene una dimensión del proceso y una dimensión de la capacidad del proceso (ver figura 2).

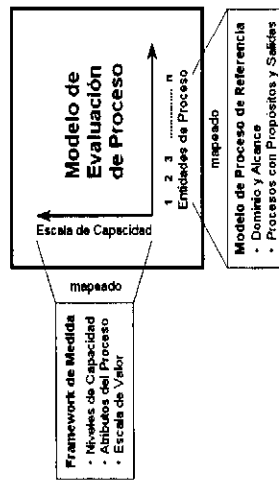


Figura 2. Vistas del modelo de evaluación de procesos

La dimensión del proceso es proporcionada por un modelo de proceso de referencia externo, el cual define un conjunto de procesos característicos con declaraciones de propósitos y resultados del proceso. La dimensión de la capacidad del proceso consiste en un framework de medida que abarca seis niveles de capacidad del proceso y sus atributos de proceso asociados.

pertinentes y adecuados del conjunto de procesos descritos en el modelo de procesos de referencia.

Light MECPDS esta basado en un conjunto de indicadores que guían los propósitos y resultados de todos los procesos dentro del modelo de evaluación de procesos. Demuestran el logro de los atributos de proceso dentro del ámbito del nivel de capacidad del modelo de evaluación. Estos indicadores son:

- Para la dimensión de la capacidad del proceso: las prácticas de gestión. Asociadas a conseguir los resultados de los atributos de proceso.
  - Para la dimensión del cumplimiento del proceso: las prácticas base. Asociadas a conseguir los resultados de los procesos definidos en el modelo de proceso referencial
- El nivel de implementación de las prácticas se evalúa a partir también de indicadores que deben ser reconocidos por la empresa para cada práctica. Pueden ser de tres tipos:
- Directos: son los productos que resultan de una actividad.
  - Indirectos: son por lo general documentos que indican que una actividad fue realizada.
  - Comentarios: son opiniones del personal relacionado con un proceso evaluado.

Light MECPDS utiliza el mapeo de los propósitos y salidas de los procesos seleccionados del modelo de proceso de referencia como indicadores de evaluación en la dimensión del cumplimiento del proceso. Además, utiliza el mapeo de los atributos del proceso del framework de medida como indicadores de evaluación en la dimensión de la capacidad del proceso.

4.1. Framework de medida de Light MECPDS

El marco de trabajo de medida de Light MECPDS esta basado en ISO/IEC 15504:2003. Se define para dos dimensiones: capacidad del proceso y cumplimiento del proceso.

La dimensión de la capacidad del proceso es definida por una escala jerárquica de tres niveles, que representan el incremento de las capacidades de los procesos de desarrollo de software. La escala queda definida de menor a mayor grado de capacidad por los siguientes niveles:

- Nivel 0. Proceso Incompleto. El proceso no esta implementado o fallan los logros de su propósito. Hay poca o ninguna evidencia de

algun logro sistemático del propósito del proceso. Hay grandes fallas que limitan o incluso impiden el cumplimiento de los objetivos y propósitos del proceso. Hay muy pocos o incluso ningún producto y/o salida identificados a lo largo del proceso.

- Nivel 1. Proceso Realizado. La implementación del proceso logra su propósito de proceso. El propósito del proceso es generalmente alcanzado, aunque éste no sea siempre planificado o controlado. Los individuos dentro de la organización reconocen que se debe llevar a cabo una acción la cual se ejecuta cuando es requerida. Existen productos generados por el proceso, utilizados para medir el logro de objetivos.

- Nivel 2. Proceso Gestionado. A la realización del proceso se le implementa una manera de gestionarlo (se planea, se monitorea y se ajusta). Sus productos de trabajo se establecen, controlan y mantienen apropiadamente. El proceso genera productos capaces de ser liberados en tiempo y bajo planes controlables. Los productos generados están alineados con determinados estándares y requerimientos. Los productos generados por procesos que se encuentran en este nivel cumplen con ciertas especificaciones puntuales de calidad respetando un cronograma y un plan.

En esta dimensión el alcanzar un nivel se demuestra por el cumplimiento de atributos de proceso. Los atributos de proceso son elementos que permiten determinar las capacidades y habilidades de un proceso. Los atributos de proceso se componen de prácticas de gestión.

Una práctica de gestión es una actividad de gestión de proceso que realiza la capacidad para realizar un proceso. Una práctica de gestión soporta la implementación o gestión de un proceso y puede ser aplicada a cualquier proceso. Las prácticas de gestión permiten su medición individual para así determinar el grado de alcance del atributo al que pertenecen y el nivel en que se encuentra el proceso en estudio. Cada uno de estos atributos, en forma individual, permite a su vez medir un aspecto específico de las capacidades y habilidades dentro de un proceso.

Cada nivel exige un grado de cumplimiento y/o un mayor número de atributos de proceso para alcanzarlo. En las tablas 1, 2 y 3 se especifican los

atributos de proceso y las prácticas de gestión asociadas a cada uno de ellos.

Cada uno de los elementos descritos anteriormente deben tener una escala específica para su medición, es así que para las prácticas de gestión y los atributos de proceso los valores se reflejan en una escala discreta compuesta por los siguientes elementos:

- **CI:** completamente implementado. Entre 86% y 100 %. Hay evidencias de una completa y sistemática aproximación, y logro total, al cumplimiento del atributo en el proceso evaluado. No hay debilidades significativas en las unidades de trabajo.
- **AI:** ampliamente implementado. Entre 51% y 85%. Hay evidencias de una aproximación sistemática, y logro significativo, al cumplimiento del atributo en el proceso evaluado. La ejecución del proceso puede variar en algunas áreas o unidades de trabajo.
- **PI:** parcialmente implementado. Entre 16% y 50%. Hay evidencia de alguna aproximación, y algún logro, al cumplimiento del atributo en el proceso evaluado. Algunos aspectos del cumplimiento del atributo pueden ser impredecibles.
- **NI:** no implementado. Entre 0% y 15%. Hay muy poco o incluso ninguna evidencia de cumplimiento del atributo definido en el proceso evaluado.

El valor de un atributo del proceso se obtiene de encontrar el promedio de los valores porcentuales de sus prácticas de gestión. Se debe considerar que cada práctica de gestión tiene el mismo peso dentro de un atributo del proceso.

La tabla 4 define el nivel de capacidad asociado a un proceso, el cual permite medir el grado de calidad de un producto de software generado por el mismo. Hay una relación entre niveles de capacidad y grado de cumplimiento de los atributos de proceso evaluado.

Se define un nivel de capacidad para cada uno de los procesos evaluados y definidos por el modelo de proceso de referencia. Pero es importante dar una visión general del estado de la capacidad de la organización. Para la evaluación del "nivel de capacidad general" de la organización se tienen en cuenta los resultados de la evaluación de los procesos asociados y definidos por el modelo de proceso de referencia (ver tabla 5).

propósito de un proceso particular y contribuye a la generación de sus salidas. Es una actividad esencial de un proceso particular.

En esta dimensión el alcanzar un proceso se demuestra por el cumplimiento de las prácticas base asociadas al proceso que se está evaluando. Las prácticas base permiten su medición individual para así determinar el grado de cumplimiento del proceso en estudio.

Para asignar un valor de implementación a las prácticas base y los procesos, se debe tener una escala específica para su medición. Estos valores están en una escala discreta compuesta por los elementos CI, AI, PI y NI, tal y como se describió anteriormente.

El valor del cumplimiento de un proceso se obtiene de hallar el promedio de los valores porcentuales de sus prácticas base, expresado este promedio en los valores definidos anteriormente. Se debe considerar que cada práctica base tiene el mismo peso dentro de un proceso específico.

Se define un valor de cumplimiento para cada uno de los procesos evaluados y definidos por el modelo de proceso de referencia. Pero es importante dar una visión general del estado del cumplimiento de los procesos de la organización.

Primero se debe obtener el valor del cumplimiento de cada una de las categorías de procesos (principales, apoyo y organizativos) definidas en el modelo de procesos de referencia. Este valor se obtiene de encontrar el promedio de los valores porcentuales de sus procesos correspondientes, expresado este promedio en términos de CI, AI, PI y NI. Se debe considerar que cada proceso tiene el mismo peso.

Para determinar el "estado general de cumplimiento del proceso" en la organización se debe tener en cuenta el valor del cumplimiento de cada una de las categorías de procesos. El valor del estado general de cumplimiento del proceso se obtiene de encontrar el promedio de los valores porcentuales de sus categorías de proceso, expresado este promedio en términos de CI, AI, PI y NI. Se debe considerar que cada categoría de proceso tiene el mismo peso.

**4.2. Modelo de procesos de referencia de Light MECPDS**

El modelo de procesos de referencia de Light MECPDS utiliza a ISO/IEC 12207/Amd.1:2002.

Tabla 1. Atributo Realización del Proceso

Id. Atributo	Descripción del atributo	Escalas
PA-1.1a	El atributo Realización del Proceso se mide en el nivel de cumplimiento de su propósito.	NI, PI, AI, CI
Nivel	Descripción de la práctica de gestión	NI, PI, AI, CI
Práctica	Identificar los productores de trabajo que son entradas del proceso.	
MP-1.1.1a	Identificar los productores de trabajo que son productores por el proceso.	
MP-1.1.2a	Identificar los productores de trabajo que son productores por el proceso.	
MP-1.1.3a	Tomar acciones para transformar los productores de trabajo de estado empotrado de salida.	

Tabla 1. Atributo Realización del Proceso

Id. Atributo	Descripción del atributo	Escalas
PA-2.1a	El atributo Gestión de la Realización se mide en el nivel de cumplimiento de su propósito.	NI, PI, AI, CI
Nivel	Descripción de la práctica de gestión	NI, PI, AI, CI
Práctica	Identificar los objetivos para la realización del proceso.	
MP-2.1.1a	Identificar los objetivos para la realización del proceso.	
MP-2.1.2a	Planear y monitorear la realización del proceso.	
MP-2.1.3a	Ajustar la realización del proceso para satisfacer los requisitos.	
MP-2.1.4a	Definir, asignar y comunicar los responsables y autoridades para realizar el proceso.	
MP-2.1.5a	Identificar, asignar, utilizar y poner a disposición los recursos necesarios para realizar el proceso.	
MP-2.1.6a	Gestionar las interfaces entre las partes involucradas para asegurar la efectiva comunicación y también la asignación clara de responsabilidades.	

Tabla 2. Atributo Gestión de la Realización

Id. Atributo	Descripción del atributo	Escalas
PA-2.2a	El atributo Gestión del Producto de Trabajo se mide en el nivel de cumplimiento de su propósito.	NI, PI, AI, CI
Nivel	Descripción de la práctica de gestión	NI, PI, AI, CI
Práctica	Definir los requisitos para los productores de trabajo del proceso.	
MP-2.2.1a	Definir los requisitos para los productores de trabajo del proceso.	
MP-2.2.2a	Definir requisitos para la documentación y control de los productos de trabajo.	
MP-2.2.3a	Identificar, documentar y controlar los productores de trabajo.	
MP-2.2.4a	Revisar de acuerdo con el plan establecido los productos de trabajo y ajustarlo como necesidad para satisfacer los requisitos.	

Tabla 3. Atributo Gestión del Producto de Trabajo

Nivel de Capacidad Organizativa	Atributos del proceso	Grado de cumplimiento esperado
Nivel 1 - Realizado	Realización del proceso   AI o CI	CI
Nivel 2 - Gestionado	Gestión de la realización   AI o CI	AI o CI
Nivel 3 - Gestionado	Gestión de los productos   AI o CI	AI o CI

Tabla 4. Cumplimiento de niveles de capacidad

Nivel de Capacidad Organizativa	Regla para alcanzar el nivel
Nivel 1 - Realizado	Si todos los procesos aplicables a la organización en el atributo del proceso PA-1.1, tienen un grado de cumplimiento esperado AI o CI entonces el nivel alcanzado por la organización es el nivel 1.
Nivel 2 - Gestionado	Si todos los procesos aplicables a la organización en los atributos del proceso PA-1.1, PA-2.1 y PA-2.2, tienen un grado de cumplimiento esperado AI o CI entonces el nivel alcanzado por la organización es el nivel 2.

Tabla 5. Determinación del nivel de capacidad general

La dimensión del cumplimiento del proceso se caracteriza por enfocarse en las características y propósitos de un proceso específico determinado y definido por el modelo de proceso de referencia.

Los procesos están compuestos por prácticas base. Una práctica base es una actividad de ingeniería de software que directamente guía el

El dominio de la norma es el suministro, desarrollo, operación y mantenimiento de productos software. Esta orientada para ser usada por una organización en el aseguramiento de la calidad de sus procesos de desarrollo de software.

El alcance de la norma es establecer un marco de referencia común para los procesos del ciclo de vida del software. Contiene procesos, actividades y tareas para aplicar durante el suministro, desarrollo, operación y mantenimiento de productos software. Los procesos son descritos en términos de lograr los propósitos y salidas.

La norma no define cómo o en qué orden se lograrán los propósitos y salidas. Los resultados serán alcanzados en una organización siguiendo prácticas detalladas para generar productos de trabajo. Estas prácticas realizadas y las características de los productos de trabajo son indicadores que demuestran si los propósitos específicos están siendo logrados. Además la norma permite a una organización definir "como" un proceso será ejecutado conservando de esta forma la flexibilidad necesaria para que los países o las organizaciones la implementen de acuerdo la cultura local o a la tecnología disponible.

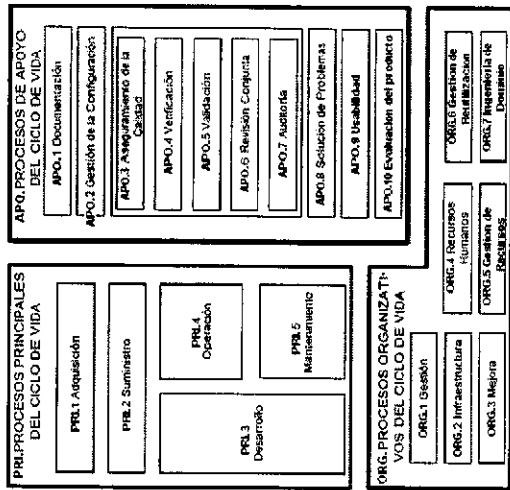


Figura 4. Modelo de Procesos de Referencia de Light MECPDS.

La estructura de los procesos de software es dividida en tres categorías que son: Principales - PRI, Apoyo - APO y Organizativos - ORG (ver figura 4). Con el fin de aligerar el modelo de

evaluación, de cada una de estas categorías se deben escoger los procesos pertinentes y aplicables que se van a evaluar en la organización a partir de los objetivos de mejora. Los elementos fundamentales del modelo de proceso de referencia son las descripciones de los procesos en términos de sus propósitos y sus resultados.

### 5. Conclusiones y trabajos futuros

En este artículo se ha presentado un modelo ligero de evaluación de la calidad de los procesos de software. Los componentes fundamentales del modelo son: un framework de medida y un modelo de procesos de referencia.

Para aligerar el modelo de evaluación se definen 3 niveles con 3 atributos de proceso (de los 9 definidos por la norma). La evaluación se aligera aproximadamente un 70%. Además, se deben escoger los procesos pertinentes y aplicables que se van a evaluar en la organización.

El marco de trabajo presentado en este artículo proporciona a las PyMES un modelo de evaluación adaptado a sus características, las cuales no disponen de los medios y recursos suficientes para la aplicación de los modelos de madurez de procesos tradicionales propuestos por el SEI o la ISO. Para la definición de Light MECPDS se han considerado las necesidades de las empresas del sector informático de Colombia, pero el marco de trabajo ha sido definido de forma general con el fin de ser aplicado a cualquier PyME del sector. En relación a otras propuestas relacionadas, Light MECPDS proporciona un modelo explícito de evaluación ligero con el fin de establecer la base necesaria para la mejora de procesos. Dicho modelo está basado en recientes normas del proceso de evaluación de ISO/IEC.

Como trabajo futuro se debe crear el instrumento de recolección de información, para la aplicación del modelo en la empresa SITIS [16] y en otras empresas piloto, para proceder a su evaluación, refinamiento y validación. Además se trabajará en una herramienta software que soporte la evaluación utilizando este marco de trabajo.

### Referencias

- [1] Mayer & Bunge Informática LTDA. Panorama de la Industria Latinoamericana de Software. Brasil. Pagina 97. 2004.

- [2] Batista J. Figueiredo A. SPI in very small team: a case with CMM. Software Process Improvement and Practice 5 (4), 243-250. 2000.

- [3] Fedesoft. II Cumbre Sectorial de entidades relacionadas con las Tecnologías de la Información. [www.fedesoft.org](http://www.fedesoft.org). 2004.

- [4] Calvo-Manzano J. Métodos de mejora del proceso de desarrollo de sistemas de información en la pequeña y mediana empresa. Tesis Doctoral. Universidad de Vigo 1999.

- [5] Maller P. Ochoa C. Silva J. Lightening the software production process in a CMM level 5 framework. JISBD. 2004.

- [6] Hareton L. and Terence Y. A Process Framework for Small Projects. Software Process Improvement and Practice 6, 67-83. 2001.

- [7] Hossein S. Natsu C. Characterizing a Software Process Maturity Model for Small Organizations. University of Nebraska at Omaha. 1997.

- [8] Hurtado J. y otros. SIMEP-SW- Sistema Integral de Mejoramiento de los Procesos de Desarrollo Software en Colombia. Colciencias. 2003.

- [9] Hurtado J. El modelo integral de mejoramiento Agile SPI. Universidad del Cauca. 2004.

- [10] ISO/IEC 12207:2002. AMENDMENT 1: Information Technology - Software Life Cycle Processes Amendment 1.

- [11] ISO/IEC 15504-2:2003. Information technology - Process assessment - Part 2: Performing an assessment.

- [12] Oktaba, H. et al. Modelo de Procesos para la Industria de Software MoProSoft Versión 1.1. Mayo 2003.

- [13] Weber K. Rocha A. Modelo de Referencia para Melhoria de Processo de Software: uma abordagem brasileira. Proc. of the QUATIC 2004, 73-78.

- [14] García F. FMESP: Marco de Trabajo Integrado para el Modelado y la Medición de los Procesos. Universidad Castilla-La Mancha. 2004.

- [15] Mas, A., Amengual E. Un nuevo modelo de evaluación de procesos de software para pymes a partir de SPICE (ISO/IEC TR-15504-5), Novática. 2001.

- [16] Soluciones informáticas integrales. Pagina disponible en: [www.sitis.info](http://www.sitis.info). 2005.

## Un entorno integrado para la reingeniería

Ignacio García-Rodríguez  
de Guzmán

Dept. de Informática, Estadística y Telemática  
Universidad Rey Juan Carlos  
28933 Móstoles (Madrid)  
[Ignacio.Garcia@urjc.es](mailto:Ignacio.Garcia@urjc.es)

Macario Polo

Dept. de Informática  
Escuela Superior de Informática  
Univ. de Castilla-La Mancha  
13071 Ciudad Real  
[Macario.Polo@uclm.es](mailto:Macario.Polo@uclm.es)

Mario Piattini

Dept. de Informática  
Escuela Superior de Informática  
Univ. de Castilla-La Mancha  
13071 Ciudad Real  
[Mario.Piattini@uclm.es](mailto:Mario.Piattini@uclm.es)

### Resumen

Este artículo describe la arquitectura de una herramienta que genera aplicaciones de tres capas (siguiendo la arquitectura multicapa) a partir de bases de datos relacionales. Para ello, la herramienta implementa un completo proceso de reingeniería en el que se utilizan diferentes metamodelos que nos ayudan a representar la información extraída para, posteriormente, generar de manera automática una aplicación que gestione la base de datos. El paso del esquema físico de la base de datos a la aplicación multicapa orientada a objetos se realiza mediante una serie de reestructuraciones y transformaciones en las que los metamodelos empleados juegan un papel fundamental. La herramienta desarrollada, además de permitirnos partir de cuatro gestores de bases de datos distintos, genera aplicaciones con un conjunto muy amplio de funcionalidades para varias plataformas.

**Palabras clave:** reingeniería, ingeniería inversa, metamodelos.

### 1. Introducción

El objetivo de la ingeniería inversa es obtener la representación de un sistema software en un mayor nivel de abstracción que el original [3]. Actualmente, la ingeniería inversa se sigue utilizando para recuperar modelos conceptuales y arquitectónicos de sistemas heredados (*legacy*), que son luego utilizados como la base para la fase de ingeniería directa, que produce una nueva versión del sistema adaptado a otros entornos y paradigmas, como el orientado a objetos [16], la computación distribuida [7], el software basado en componentes [1], etc.

De acuerdo con [2], el proceso completo de ingeniería inversa y directa se corresponde con la idea de "reingeniería". La reingeniería puede entenderse como la composición de un conjunto de funciones de transformación que operan sobre diferentes conjuntos, incrementando o decrementando el nivel de abstracción de un producto software. Adicionalmente, puede haber algunas fases intermedias de reestructuración, que cambian algunos de los productos software sin salir del nivel de abstracción.

Por lo general el conjunto inicial comienza desde el código fuente de los programas, siendo los modelos de clases u otros tipos de diagramas los que forman el conjunto final de esta primera transformación. No obstante, ha habido también muchos trabajos que han aplicado ingeniería inversa a otros tipos de productos, como las bases de datos. El objetivo de muchos de ellos es obtener un diagrama que represente el posible esquema conceptual usado durante el desarrollo inicial de la base de datos, con lo que el producto obtenido de la ingeniería inversa suele ser un diagrama entidad-relación (ER) o entidad-relación extendido (EER) [8, 17].

Con esta herramienta, pretendemos generar automáticamente aplicaciones capaces de gestionar bases de datos relacionales con un propósito general (aunque muy fácilmente adaptables a propósitos específicos), por lo que la obtención de un diagrama de clases que exprese la misma semántica que el diagrama ER o EER, nos facilitará esta tarea.

Mediante un diagrama de clases podemos expresar lo mismo que mediante un diagrama ER o EER, y además obtenemos la ventaja de disponer de una representación más cercana a la aplicación objetivo, con lo que la generación de código se hace mucho más sencilla. De hecho, este diagrama de clases también es un punto de partida excelente para generar nuevas versiones de la base de datos.