

# UNA PROPUESTA DE INTRODUCCIÓN DE LAS REVISIONES EN EL PROCESO DE DESARROLLO DE SOFTWARE

Martha Dunia Delgado Dapena<sup>1</sup>, Sofía Álvarez Cárdenas<sup>2</sup> y Alejandro Rosete Suárez<sup>3</sup>  
Centro de Estudios de Ingeniería de Sistemas (CEIS),  
Instituto Superior Politécnico “José Antonio Echeverría” (ISPJAE), Ciudad de La Habana, Cuba

## RESUMEN

El resultado final de un proyecto de software es un producto que toma forma a lo largo del desarrollo del proyecto. La calidad del producto final, está estrechamente ligada a la calidad del Proceso de Desarrollo de Software, entre otros aspectos. Establecer un adecuado Sistema de Aseguramiento de Calidad que contribuya a detectar los defectos en las etapas tempranas del proceso de desarrollo de Software contribuye considerablemente a elevar la calidad de los productos resultantes. Una parte importante de este Sistema lo constituyen las Revisiones, que se realizan en los diferentes momentos del ciclo de desarrollo del proyecto. Con ellas se detectan defectos que pueden ser corregidos oportunamente, de forma tal que el producto final tenga la calidad necesaria. En este trabajo se expone una propuesta para introducir las revisiones en la Pequeña y Mediana Empresa (PYME) desarrolladora de software, aplicable a las condiciones de la industria nacional. Se describe un procedimiento para su introducción paulatina, así como las métricas correspondientes, de forma tal que la empresa pueda transitar gradualmente de la situación actual a un nivel superior de organización del Proceso de Desarrollo de Software, que permita obtener productos con mayor calidad.

## ABSTRACT

The last result of a software project is a product that takes shape through all the development of the project. The quality of the last product is closely related to the quality of the Software Development Process, among other aspects. Setting the proper Quality Assurance System that contributes to detect the defects in the early stages of the Software Development Process, which helps to increase the quality of the outgoing products. Important sorts of this system are the Revisions, that are made in the different moments of the cycle of development of the project. It detected defects that can be corrected appropriately, in such way that the final product has the necessary quality. In this paper, a proposal is showed, to introduce all the Revisions of the enterprise, which develops the software. A procedure is described to introduce it step by step, so are the corresponding metrics, so the enterprise can function gradually from the current situation to a high level of organization of the Software Development Process, which allows getting products with a higher quality.

**Key words:** revisions, inspections, Software Quality Assurance, Software Engineering.

## 1. INTRODUCCIÓN

En el mundo actual en el que se necesita obtener productos con alta calidad, que integran variadas tecnologías y cuyo tiempo de elaboración sea mínimo, se impone mejorar el proceso de desarrollo de software, incrementando la productividad de los equipos involucrados en el desarrollo de los proyectos, Humphrey (1997).

El desarrollo de una Industria Nacional de Software es una tarea de gran prioridad para el estado cubano, debido a la alta perspectiva económica que posee, así como para el aseguramiento de un grupo importante de actividades del país, Moreno (2003).

Varios trabajos reflejan la importancia de establecer el proceso de revisión en las empresas de software sustentado en el argumento de que las dos terceras partes de los defectos de los sistemas son el resultado de errores cometidos en etapas tempranas del desarrollo del proyecto y sólo una tercera parte son el resultado de errores cometidos en etapas avanzadas, por lo que se hace necesario prevenir los defectos o detectarlos en las etapas tempranas, Fox (1997), Reynolds (1995), Markus (2000).

---

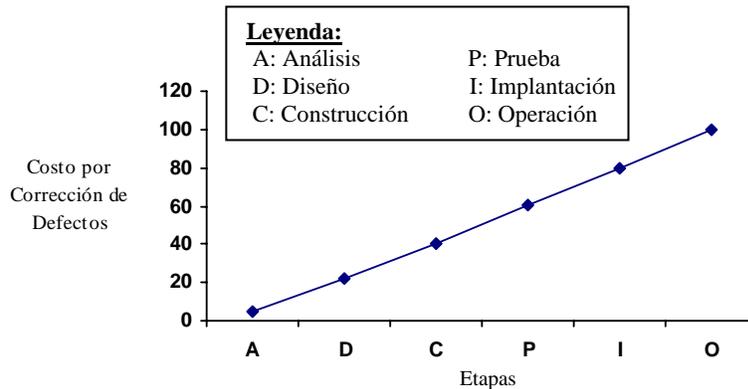
E-mail: <sup>1</sup>marta@ceis.cujae.edu.cu

<sup>2</sup>sofia@ceis.cujae.edu.cu

<sup>3</sup>arosete@ceis.cujae.edu.cu

La calidad del software puede medirse después de elaborado el producto, pero esto puede resultar muy costoso si se detectan problemas derivados de imperfecciones en el diseño, por lo que es imprescindible tener en cuenta tanto la obtención de la calidad como su control durante todas las etapas del ciclo de vida del software, Oktaba (2002).

En la Figura 1 se muestra un gráfico presentado a partir de un estudio realizado en empresas productoras de software en el mundo, que muestra el costo por concepto de corrección de defectos en las diferentes etapas del ciclo de vida del proyecto y éste evidencia que mientras más tarde se detectan los defectos más costosa es su corrección, Reynolds (1995). Más que recurrir a una revisión final, se debe atender al proceso mismo, detectando los defectos y poniendo las acciones correctivas correspondientes para prevenirlos en adelante.



**Figura 1.** Costo de corregir defectos en cada fase del ciclo de vida del proyecto.

Para detectar a tiempo los defectos y poder eliminarlos lo más conveniente sería establecer un adecuado Plan de Revisiones, donde queden planificados los momentos en que se realizarán las Revisiones durante el Proceso de Desarrollo, considerando en esta planificación tanto las ejecutadas por los propios desarrolladores, como las realizadas por los especialistas no involucrados en el desarrollo del proyecto.

El Modelo de Madurez de las Capacidades (CMM) propuesto por el Instituto de Ingeniería de Software (SEI) -por sus siglas en inglés-, de la Universidad Carnegie Mellon en los Estados Unidos, que ha sido adoptado por numerosas empresas desarrolladoras de software en el mundo, contempla la introducción de Revisiones como un requisito para pertenecer al segundo nivel de CMM, que avala la calidad del proceso de desarrollo en sus diferentes estadios en el perfeccionamiento de la calidad del producto final, Paul (1999), SEI (2001).

La mayor parte de los trabajos en este sentido están planteados para empresas grandes, sin embargo según datos publicados por el SEI alrededor de un 10% de las empresas en el mundo tienen 25 o menos empleados, lo que demuestra que un porcentaje significativo de las empresas a nivel mundial pertenecen a la Pequeña y Mediana Empresa (PYME). Por tanto, se hace necesario definir modelos y herramientas para este tipo de empresas, SEI (2000).

En estudios realizados por el Centro de Estudios de Ingeniería de Sistemas (CEIS) del ISPJAE en empresas nacionales se detectaron problemas entre los que se encuentran: los resultados alcanzados no cubren las expectativas, la productividad es baja, la cantidad real de recursos a consumir -en tiempo principalmente- es casi impredecible, el trabajo realizado casi nunca tiene la calidad y profesionalidad requerida, los proyectos sufren atrasos excesivos y no existen antecedentes de datos históricos sobre la calidad de los productos que han sido elaborados, Febles (2001).

Esencialmente, el Aseguramiento de la Calidad del Software consiste en la Revisión de los productos y su documentación relacionada, para verificar su cobertura, corrección, confiabilidad y facilidad de mantenimiento, Kan (1995). Y, por supuesto, incluye la garantía de que un sistema cumpla las especificaciones y los requisitos para el uso y desempeño deseados.

## 2. REVISIONES A PROYECTOS DE SOFTWARE

Las Revisiones son actividades de control de calidad, que permiten detectar defectos en los proyectos de software. Las Revisiones pueden ser de dos tipos, tal y como se muestra en la Figura 2, dinámicas y estáticas. Las primeras son las que detectan los defectos ejecutando el software, fundamentalmente son las ejecutadas en la fase de prueba del proyecto. Las segundas son visuales y se realizan sin necesidad de que

el software esté ejecutándose. Ambas son de suma importancia y una combinación adecuada puede detectar gran cantidad de defectos y por tanto mejorar la calidad del producto final.



**Figura 2.** Tipos de Revisiones.

Algunos autores describen las fases de forma general, así como los roles de sus participantes, Schulmeyer (1997), mientras que otros se concentran en abordar el Equipo de desarrollo y hablar de las Revisiones que realiza el propio equipo de desarrollo, estableciendo los documentos para registrar la información resultante, pero sin detallar el procedimiento a seguir, Humphrey (1997). Aunque los diferentes autores coinciden en la importancia de las revisiones en etapas tempranas, sus propuestas se limitan a sugerir su introducción sin definir el procedimiento a seguir, excepto para las inspecciones, en las que se ha definido, además de las fases y los roles con sus respectivas responsabilidades, el procedimiento de ejecución de forma detallada.

No obstante, en la bibliografía consultada no se ha encontrado un procedimiento para introducir las Revisiones en la PYME de forma paulatina, de modo que se pueda transitar hacia un estado de mayor calidad en el que se aplique el procedimiento en detalle, Quintanilla (2002).

En la situación de la industria nacional, en la que no existe disciplina en la aplicación de estas técnicas es prácticamente imposible pensar en introducirlo todo "de golpe", por lo que se hace necesaria una propuesta de procedimiento para introducir las Revisiones en la industria Nacional, teniendo en cuenta las condiciones actuales. A continuación se hace una propuesta que puede ser empleada para tales propósitos.

### **3. PROPUESTA DE PROCEDIMIENTO PARA INTRODUCIR LAS REVISIONES EN LA PYME**

La Pequeña y Mediana Empresa (PYME) no cuenta con suficiente personal como para asumir las revisiones con rigurosidad, pues apenas alcanzan para desarrollar los proyectos que se les asignan. En estudios realizados por el Centro de Estudios de Ingeniería de Sistemas (CEIS) en empresas cubanas se detectaron problemas relacionados con, Álvarez (2000):

- El personal disponible en estas empresas es aún escaso aunque tiene un alto nivel de preparación.
- Existe una gran desorganización en las empresas lo que no permite aplicar técnicas, modelos o estándares que ayudarían al desarrollo de ésta.
- Existen pocos clientes y no se ha creado una disciplina para el intercambio con ellos.
- El mercado externo es aún muy pobre, pues prácticamente no se obtienen ingresos por concepto de exportación de software.
- No existe una cultura de producción de software bajo parámetros de terminación y calidad, donde se actúe bajo conceptos y estudios técnicamente fundamentados por equipos multidisciplinares y competentes dirigidos a la creación de un producto orientado a determinado mercado.

- Hay mala calidad en gran parte del software que se produce en el país y es indispensable la solución de este problema lo más brevemente posible, pues la calidad del producto que desarrollan las empresas nacionales es clave para mejorar su competitividad, y teniendo la calidad del producto como elemento distintivo, éstas pueden encontrar nuevos mercados.

Los resultados de este estudio arrojan la necesidad de establecer políticas de calidad en las organizaciones con el fin de mejorar la productividad de las empresas y lograr una adecuada satisfacción de los clientes tanto en el mercado nacional como en el internacional.

Establecer un adecuado Sistema de Aseguramiento de la Calidad contribuye considerablemente a elevar ésta en el o los productos resultantes del proyecto de software. Una parte importante de este sistema lo constituyen las Revisiones, que se realizan en los diferentes momentos del Proceso de Desarrollo del proyecto. Con ellas se detectan errores que van desde la omisión de algún requisito, en las etapas tempranas del desarrollo del proyecto, hasta la detección de errores en el código de algún producto de software asociado con éste.

En un diagnóstico realizado por los autores, en Diciembre del 2003, a dieciocho empresas desarrolladoras de software en el país se detectó por un lado que a pesar de la existencia en las organizaciones de directivas de calidad, donde se incluyen las revisiones, y de la disponibilidad de recursos para llevarlas a cabo, no se elaboran y aprueban planes de revisiones y auditorías para los proyectos, solo se realizan pruebas al finalizar el producto y no se registran datos sobre los defectos encontrados que permitan tener una medida de la calidad de los productos y del proceso de detección de los mismos. Algunos de los resultados se muestran en las Figuras 3, 4 y 5.

Los resultados obtenidos evidencian la necesidad de hacer propuestas referentes a la planificación, ejecución y medición del proceso de revisiones. A continuación se expone una propuesta de procedimiento para introducir las Revisiones en las PYME, considerando cada uno de estos elementos.

### 3.1. Propuesta de procedimiento para introducir las Revisiones en la PYME

La propuesta que se expone a continuación es el resultado de un conjunto de esfuerzos realizados por el CEIS para introducir las Revisiones y otros procesos en la PYME cubana, Delgado (2003), Febles (2004). El Modelo de Proceso de Revisiones, consta de un Sistema de Procedimientos, Métricas y Herramientas para introducir paulatinamente, como se muestra en la Figura 6, adecuándolo a las características de cada empresa en el país, considerando la insistencia de varios autores en cada modelo debe ser adecuado a las condiciones propias de

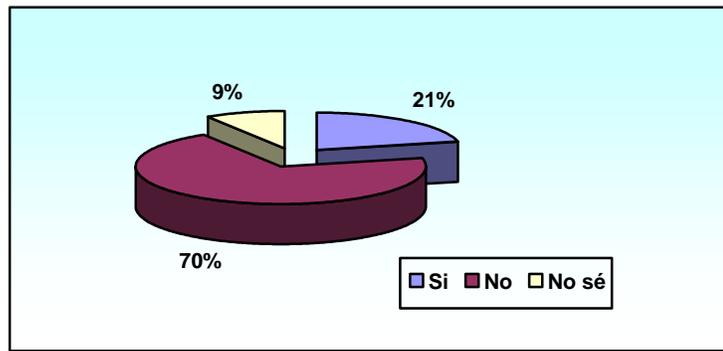


Figura 3. Elaboración del Plan de Revisiones y Auditorías para cada proyecto.

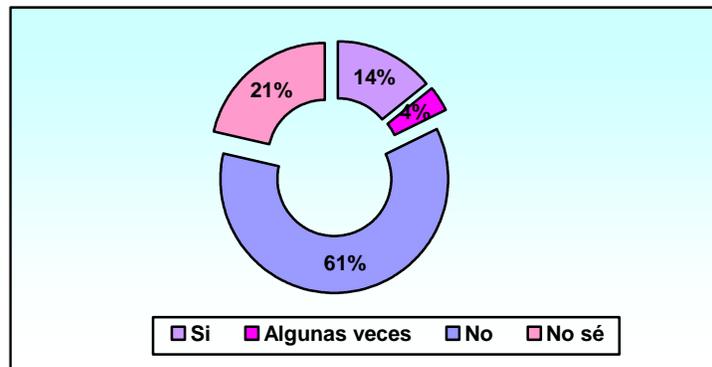


Figura 4. Utilización de métricas para defectos y revisiones.

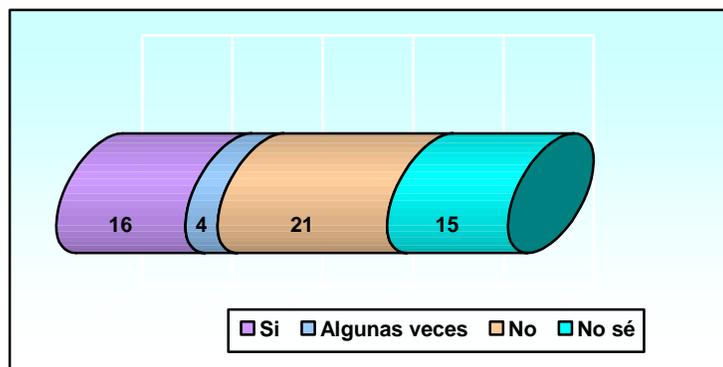
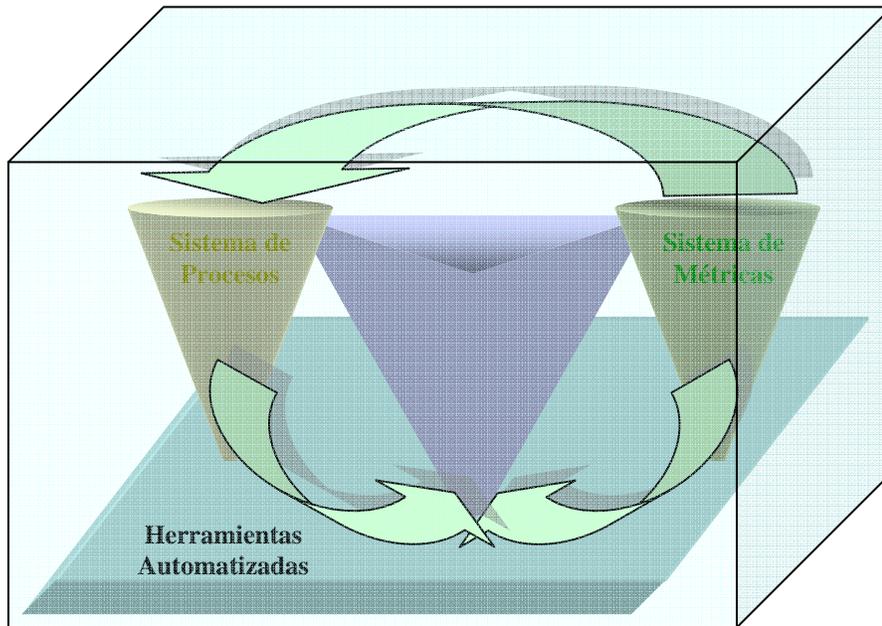


Figura 5. Existencia de guías para revisión o prueba de proyectos.

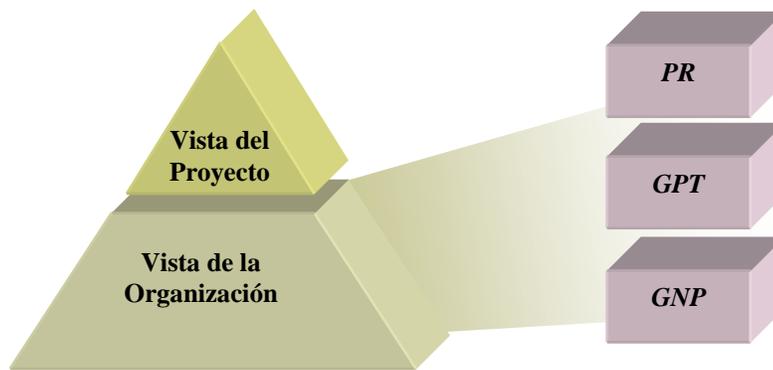
cada empresa, Reo (2002), Oktaba (2003), Quintanilla (2002). Se definen los roles del proceso de acuerdo a las características de la empresa desarrolladora de software en Cuba y se realizan talleres con las organizaciones interesadas para adecuar el Modelo a sus particularidades.



**Figura 6.** Modelo de Proceso de Revisiones para la PYME.

El Sistema de Procesos propuesto por los autores tiene dos vistas, una a nivel de organización y otra a nivel de proyecto, ambas con un conjunto de procesos mostrados en la Figuras 7 y 8 respectivamente.

A continuación se describen cada uno de los procesos y las actividades contenidas en cada uno de ellos.



**Figura 7.** Sistemas de Procesos: Vista de la Organización.

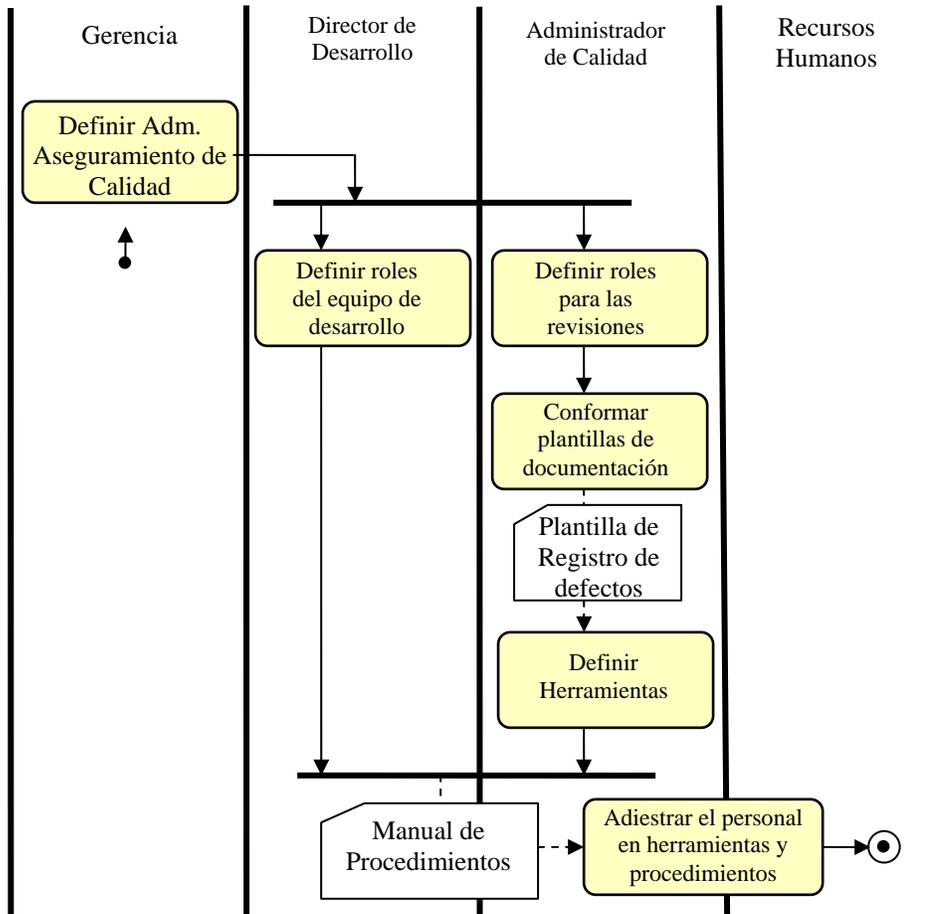


**Figura 8.** Sistema de Procesos: Vista del Proyecto.

## VISTA DE LA ORGANIZACIÓN

### Preparación de las Revisiones (PR)

En esta etapa la empresa deberá trazar pautas para el desarrollo de las Revisiones, los roles que se ejecutarán, así como las políticas generales de Aseguramiento de Calidad. Las actividades propuestas son las siguientes (ver Figura 9):



**Figura 9.** Diagrama de actividades del Proceso de Preparación.

1. Definir un Administrador de Aseguramiento de la Calidad, encargado de establecer las políticas y velar por la calidad de los productos a nivel de empresa,
2. Definir roles adecuados para el Grupo de Desarrollo de cada Proyecto de software, entre los que deben estar al menos: Líder del Proyecto, Desarrollador y Director de Calidad, Humphrey (1997). Aunque los roles propuestos por Humphrey en el Proceso de Software en Equipo (TSP) -por sus siglas en inglés- son cinco, es necesario considerar las características de las PYME en las que los proyectos generalmente son desarrollados por equipos con entre dos y cuatro individuos, por lo que esta propuesta considera solo tres roles y las responsabilidades se distribuyen como se muestran en la Tabla 1.

**Tabla 1.** Adecuación de Roles para las PYME considerando los roles propuestos en TSP.

Roles Propuestos	Roles de TSP
Líder de Proyecto	Líder del Equipo
	Director de Planificación
Director de Desarrollo	Director de Desarrollo
	Director de Apoyo
Director de Calidad	Director de Calidad

3. Definir roles para las revisiones, que dependerán del tipo de revisión que se desarrolle.

La inspección es ejecutada por un Equipo de Inspección, en el que participan: el *Moderador*, que distribuye el trabajo a los inspectores y actúa como facilitador durante el desarrollo de la inspección; el *Lector*, que se encarga de llevar todos los registros de la Inspección y ordenar la información para que sea empleada adecuadamente en la Reunión de Inspección; el *Inspector*, que revisa el producto para encontrar los defectos y si lo entiende necesario puede proponer posibles soluciones y el *Autor*, que responde por el desarrollo del producto que se revisa.

Las pruebas son ejecutadas por un Equipo de prueba en el que participan, según el Proceso Unificado de Rational, Racional (2001), los roles siguientes: diseñador de pruebas, ingeniero de componentes, ingeniero de pruebas de integración e ingeniero de pruebas de sistema. Por las características de escaso personal que tiene la PYME se propone trabajar con el diseñador de pruebas y el ingeniero en componentes y que ellos mismos realicen las pruebas de integración y sistema.

4. Establecer los documentos para registrar el resultado de las revisiones y los involucrados dentro de la organización que recibirán estos informes,
5. Definir las herramientas para la ejecución de las revisiones,
6. Adiestrar al personal en el uso de las herramientas y procedimientos establecidos.

### **Gestión de productos terminados (GPT)**

En este proceso se revisan los productos terminados por la empresa con el objetivo de recoger la mayor cantidad de información posible sobre los defectos encontrados. La empresa debe establecer políticas que le permitan registrar la información de los defectos encontrados en los productos que ya están en manos de los clientes o que se encuentran en las fases de implantación y explotación. El objetivo fundamental de este proceso es contar con la información del estado y calidad de los productos desarrollados por la empresa antes de introducir el modelo, de forma que la empresa pueda trazar acciones preventivas, para ello se ejecutan las actividades siguientes:

1. Registrar defectos reportados por los clientes, para el caso de productos en fase de explotación u operación,
2. Registrar defectos reportados por el personal que implanta el sistema, para el caso de los productos en fase de implantación,
3. Llevar métricas para ambos tipos de productos y trazar acciones preventivas.

### **Gestión de nuevos productos y productos en desarrollo (GNP)**

La organización debe definir un conjunto de políticas con respecto a las Revisiones, que deberán ser seguidas por desarrolladores e inspectores a lo largo del proceso de desarrollo de software y durante la ejecución de las Revisiones. Se definirán los momentos, tipos de Revisión y listas de chequeo o listas de comprobación generales para cada tipo de proyecto, que son aquellas listas cuyos elementos consisten en parámetros o preguntas a considerar en la revisión y que pueden ser considerados defectos de estar presentes en el proyecto. Además es importante que no deje de declararse explícitamente con que frecuencia se revisará la línea base del proyecto que pasará a formar parte del repositorio de elementos de configuración, como una versión **estable y revisada** de los elementos de configuración correspondientes.

## **VISTA DEL PROYECTO**

### **Proceso de Elaboración del plan de revisiones y auditorías (EPR)**

Debe tener en cuenta las políticas definidas por la organización y fijar cuantas revisiones adicionales se consideren necesarias, cuidando no retrasar innecesariamente la ejecución del proyecto. El responsable de elaborar el plan es el Director de Calidad, conjuntamente con el Líder de Proyecto, posteriormente debe efectuarse la aprobación del plan por parte del administrador de aseguramiento de la calidad de la empresa.

Se recomienda realizar al menos las siguientes revisiones y auditorías durante el desarrollo del proyecto, Delgado (2003a):

- Inspecciones del Colectivo: ejecutadas por el propio equipo de desarrollo, en este caso se recomiendan las revisiones de los requisitos, la arquitectura, el Diseño y los ciclos de desarrollo, Jacobson (2000). Por otro lado, las revisiones de la arquitectura y el diseño son las que revisan el diseño preliminar y crítico respectivamente, que persiguen el objetivo de detectar defectos antes de pasar a la fase de codificación, y por último las revisiones al finalizar cada ciclo de desarrollo detectan defectos presentes en los componentes y artefactos del proyecto desarrollados en cada uno de los ciclos,
- Auditoría de la configuración: ejecutadas por personal ajeno al desarrollo del proyecto, se deben ejecutar antes de incluir la línea base del proyecto en el repositorio de elementos de configuración, para garantizar la estabilidad de los elementos de configuración. Además debe realizarse al terminar cada versión del software. El objetivo fundamental de estas revisiones es detectar defectos que los propios desarrolladores no sean capaces de detectar porque estén relacionados con la lógica de pensamiento seguida por ellos en el desarrollo del proyecto, en ella se verifica que todos los requisitos han sido cumplidos y que el software y su documentación están completos y listos para entregar,
- Revisiones dinámicas, en las que se deben ejecutar las actividades de evaluación y prueba del software, considerando en cada momento el tipo de prueba que se requiere.

### **Detección y seguimiento de defectos (DSD)**

Las actividades del proceso varían dependiendo del tipo de revisión que se efectúe, si se trata de una inspección se comenzará por designar un moderador, tal y como se muestra en la Figura 10, encargado de conducir todo el proceso hasta obtener el reporte final de la inspección y entregar al administrador de aseguramiento de calidad que se encargará de emitir las peticiones de cambio correspondientes, Febles (2004), si fuese necesaria una reinspección. En el caso de una revisión dinámica las actividades son las siguientes:

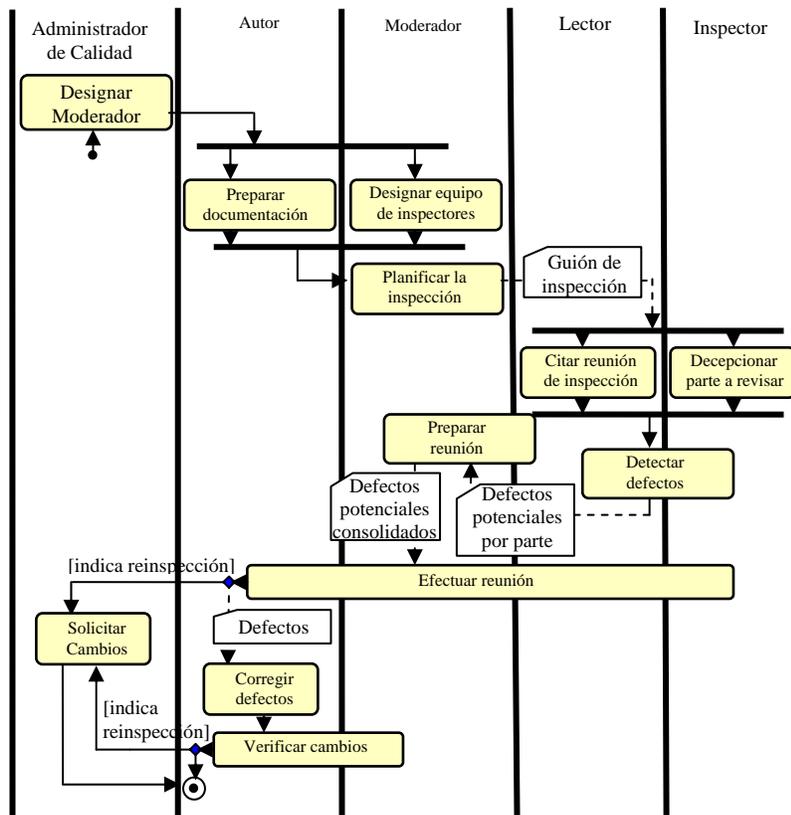
1. Designar el equipo de prueba o evaluación del software,
2. Confeccionar casos de prueba y procedimientos de prueba o en su defecto las listas de comprobación correspondientes,
3. Detectar y registrar los defectos a partir de la ejecución del software,
4. Notificar al administrador de aseguramiento de calidad y director de calidad del proyecto,

Para la ejecución de las auditorías de configuración, deben considerarse los siguientes pasos:

- Designar el auditor responsable o solicitar los servicios de una agencia consultora, según sea el caso, y el grupo de auditores involucrados,
- Entregar a los auditores el plan de configuración del proyecto,
- Revisar que todos los requisitos hayan sido cumplidos y registrar las inconformidades en el registro de auditoría,
- Revisar que la documentación esté completa y que los elementos de configuración estén listos para ser entregados,
- Reunirse con el equipo de desarrollo del proyecto para informar los resultados de la auditoría,
- Generar las peticiones de cambio correspondientes por parte del director de desarrollo del proyecto, si fuese necesario.

### **Prevención de defectos (PD)**

El proceso comienza con la definición de las métricas para analizar el comportamiento de los defectos en el proyecto y la comparación con otros proyectos desarrollados en la empresa y el objetivo fundamental es tomar acciones preventivas que permitan mejorar el proceso de revisiones y el propio proceso de desarrollo de software, a partir del análisis de las métricas. Para ello se propone un Sistema de Métricas conformado por un total de 12, que debe considerar cada empresa y adecuar a sus condiciones particulares. Las métricas que se presentan han sido tomadas del libro "Metrics and Models in Software Quality Engineering", Kan (1995), adecuándolas a las condiciones de las PYME y considerando otros tipos de revisión, pues Kan en su libro únicamente considera las inspecciones.



**Figura 10.** Diagrama de actividades de ejecución de la inspección.

En la Tabla 2 se hace una propuesta de utilización de métricas para evaluar el cumplimiento de algunas de las metas de la organización, vinculadas específicamente con las revisiones.

**Tabla 2.** Propuesta de utilización de métricas provenientes de las Inspecciones.

Meta	Pregunta	Métrica
Planificación	¿Cuánto cuesta el proceso de revisión?	ELC
	¿Cuánto tiempo consume el proceso de revisión?	ELC EIP
Control y Seguimiento	¿Cuál es la calidad del software inspeccionado?	DELC
		PR
		RPP
		DD
	¿En qué medida el personal técnico sigue los procedimientos establecidos para las revisiones?	PR
		RPP PLCR PRI EIP
¿Cuál es el estado del proceso de inspección?	TLCR	
	EIP	
Mejoramiento	¿Cuán efectivo es el proceso de revisión?	EED
		EED <sub>i,j</sub>
		DELC
		PR
		RPP
		PLCR
	¿Cuál es la productividad del proceso de revisión?	DD
		EDE
		PR
		RPP PLCR

Descripción de cada una de las métricas:

1. Efectividad de eliminar los defectos en una Revisión:

$$EED = \frac{DE_i}{DL} * 100$$

$$DL = DE_i + DEP$$

$$DEP = \sum_{k=i+1}^n DE_k$$

DE: cantidad de defectos detectados durante la revisión,

DEP: cantidad de defectos encontrados posterior a la revisión, es decir la cantidad de defecto encontrados en las n-i restantes revisiones que se indican en el plan de revisiones y auditorías del proyecto. También puede calcularse este valor considerando los defectos detectados en las revisiones efectuadas hasta el momento (k = m, con i < m < n) en que se desea analizar la métrica, pero serán resultados parciales que pueden cambiar al finalizar el producto.

DL: cantidad total de defectos presentes en el producto, cuando éste ha sido terminado y se entrega al cliente para su operación.

Si se representan en un gráfico los valores resultantes de la métrica de efectividad se puede analizar qué revisiones de las realizadas al Proyecto de Software resultan poco efectivas y así valorar la posibilidad de incluirla o no en la misma fase del Desarrollo del Proyecto o en otros cuyas características sean similares al analizado o tomar cualquier otra decisión que contribuya al mejoramiento del Proceso de Revisiones.

Ahora bien, esta medida da una vista global de la efectividad de la revisión, pero en ocasiones no basta con esta información y es necesario profundizar para conocer cuáles tipos de errores no han sido detectados y que por tanto conspiran contra la efectividad de dicha Revisión, para ello se propone la métrica siguiente.

2. Efectividad de eliminar los defectos de la fase j en la revisión i.

Permite a los directivos conocer la efectividad de las revisiones en cuanto a la cantidad de defectos que pertenecen a una fase y que son encontrados oportunamente. Por tanto, se puede analizar la cantidad de defectos, por ejemplo de la fase de requisitos, que se han propagado hasta la implantación del sistema o hasta cualquier otra fase de desarrollo del proyecto.

$$EED_{i,j} = \frac{DE_{i,j}}{DL_j} * 100$$

DE<sub>i,j</sub>: cantidad de defectos detectados durante la revisión i, correspondientes a la fase j.

DL<sub>j</sub>: cantidad total de defectos presentes en el producto correspondientes a la fase j.

El comportamiento de estas dos primeras métricas puede ser representado en un único modelo que de una idea global de la eficiencia de las Revisiones y de cada una de las fases consideradas en ellas. Por ejemplo, se pudiera plotear los valores de la efectividad de diferentes revisiones realizadas a un proyecto y compararlas entre sí.

3. Densidad de defectos.

$$DD = \frac{DT}{TP}$$

DT: cantidad total de defectos encontrados en el producto.

TP: tamaño del producto, puede ser estimado en líneas de código (en miles) o en puntos de función.

Da una medida de la proporción de defectos del producto con respecto a su tamaño. De la forma en que está definida debe ser evaluada al finalizar el producto y puede ser aprovechada como experiencia para proyectos futuros, no obstante en las etapas tempranas de desarrollo del proyecto esta métrica puede ser utilizada considerando la estimación preliminar de líneas de código o puntos de función que haya sido considerada en la planificación del proyecto.

La densidad de defectos de cada uno de los proyectos revisados en la empresa da una idea de la calidad de los productos que se desarrollan en esta y de la mejora continua del proceso de revisiones al transcurrir un periodo de tiempo determinado.

4. Porcentaje de reinspección (solo para inspecciones).

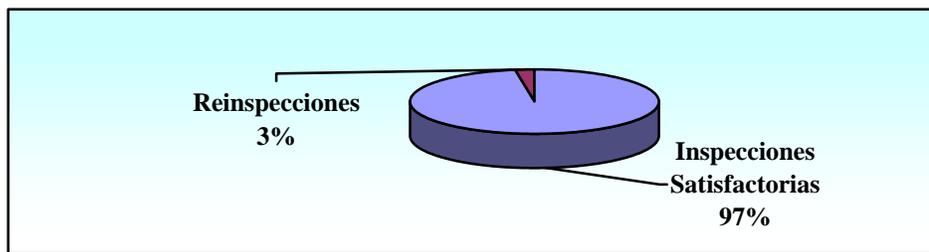
$$PRI = \frac{CR}{CI} * 100$$

CR: cantidad de disposiciones de reinspecciones, que son indicadas por el equipo de inspección si entiende que los defectos son demasiados o muy complejos como para considerar satisfactoria la inspección al producto que se realiza. En estos casos se evalúa de mal el producto inspeccionado y se indica realizar nuevamente una revisión cuando hayan sido corregidos los defectos.

CI: cantidad total de inspecciones.

Esta métrica puede ser considerada tanto para análisis del porcentaje de reinspección de cada proyecto, como para considerar el porcentaje de reinspección del proceso en general, considerando todos los proyectos que han sido inspeccionados.

En la Figura 11 se muestra el comportamiento de las reinspecciones con respecto a las inspecciones realizadas al proyecto RevisionCASE elaborado en el CRIS. Se observa que el proyecto analizado, no tiene grandes problemas, pues en el 97% han sido consideradas satisfactorias. Un análisis más profundo y detallado puede ser realizado considerando otras métricas, descritas en lo adelante.



**Figura 11.** Datos de reinspecciones al proyecto RevisionCASE.

5. Cantidad total de líneas revisadas.

$$TLCR = \sum_{k=1}^n LCR_k$$

LCR<sub>k</sub>: cantidad de líneas de código revisadas en la revisión k.

n: número total de revisiones.

6. Promedio de líneas revisadas en cada revisión.

$$PLCR = \frac{TLCR}{n}$$

Con la información que se obtiene de esta métrica se puede realizar análisis sobre la amplitud de las revisiones, pues da una idea de cuán exhaustivamente se revisó cada proyecto. La comparación entre el promedio de líneas inspeccionadas en los diferentes proyectos se muestra en el gráfico de la Figura 12.

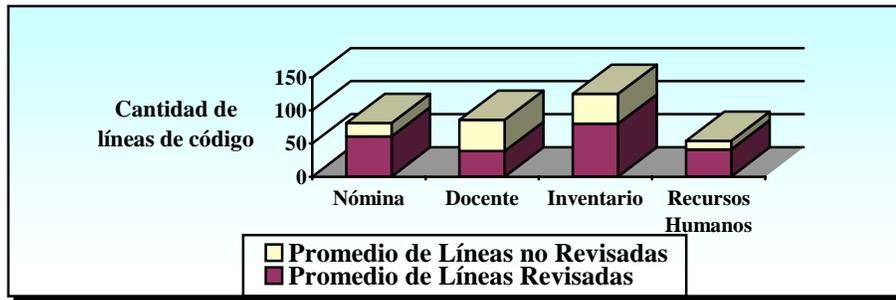


Figura 12. Estimado de líneas revisadas por proyecto.

7. Productividad de la revisión promedio.

$$PR = \frac{TLCR}{\sum_{k=1}^n TDurac_k}$$

$TDurac_k$ : tiempo de duración de la revisión k.

Permite hacer un análisis de la productividad de las revisiones al proyecto, puede ser utilizada para comparar el desempeño de las revisiones realizadas a los diferentes proyectos (Ver Figura 13).

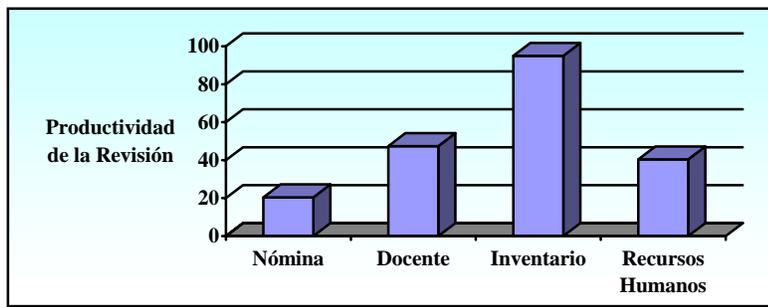


Figura 13. Productividad de la revisión por proyecto.

8. Razón de preparación promedio de los inspectores.

$$RPP = \frac{TLCR}{\sum_{k=1}^n TPrep_k}$$

$TPrep_k$ : tiempo de preparación de la revisión k.

$TInsp_k$ : total de inspectores que participaron en la revisión k.

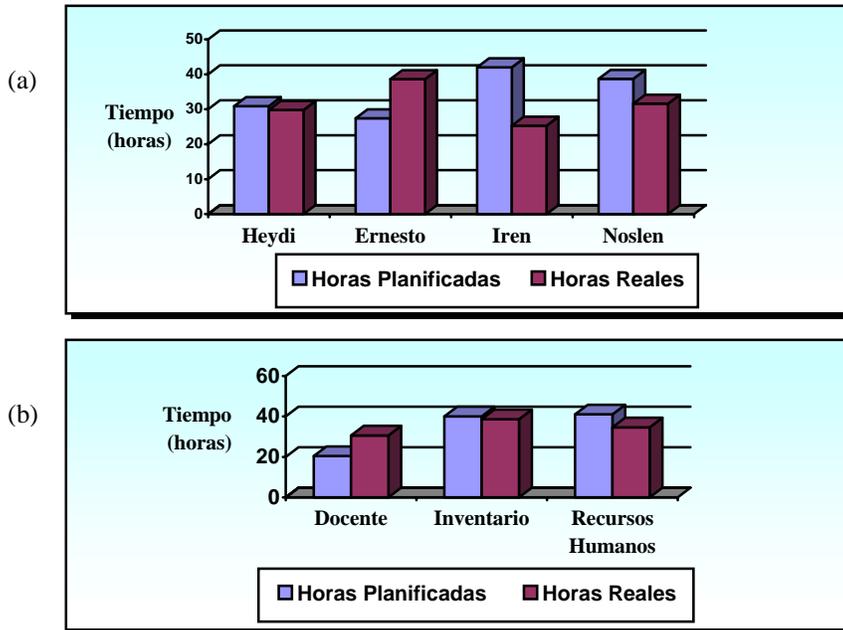
La Razón de preparación promedio de los inspectores en una revisión es un indicador de la cantidad de líneas que inspeccionó un inspector del total de líneas inspeccionadas durante la fase de preparación, de las revisiones que se le realizaron al proyecto, y puede emplearse para la planificación de futuras revisiones a proyectos con características similares al analizado. La fase de preparación es la fase de detección de defectos.

9. Eficiencia del inspector en la fase de preparación (EIP).

Esta medida se puede obtener a partir de la información reflejada en el Registro de Seguimiento de la Revisión y permite comparar lo planificado con lo real. Además se puede hacer análisis sobre el desempeño de los inspectores y saber qué inspectores son los más eficientes.

El modelo de la Figura 14a refleja la diferencia entre lo planificado y lo real en cuanto al tiempo de preparación invertido por los inspectores en la fase de preparación.

Esta métrica también puede ser analizada para los proyectos, de forma que se tenga en cuenta para la planificación de proyectos que se desarrollen con posterioridad, en cuyo caso el modelo reflejaría la relación de horas planificadas y reales para cada proyecto, como se muestra en la Figura 14b.



**Figura 14.** Cumplimiento del plan de Revisión.

(a) Tiempo real y planificado de preparación para cada inspector.

(b) Cumplimiento de la planificación de la fase de preparación de la Revisión de cada Proyecto.

10. Esfuerzo promedio por líneas de código fuente.

$$ELC = \frac{\sum_{k=1}^n Esf_k}{TLCR}$$

$$Esf_k = TPr ep_k + TPart_k * TDurac_k + TRetrab_k$$

$Esf_k$ : esfuerzo de la revisión k.

$TPart_k$ : total de participantes de la revisión k.

$TRetrab_k$ : tiempo de retrabajo o de corrección de defectos de la revisión k.

11. Esfuerzo promedio por defectos detectados.

$$EDE = \frac{\sum_{k=1}^n Esf_k}{\sum_{k=1}^n DE_k}$$

12. Promedio de defectos detectados por líneas de código fuente.

$$DELCL = \frac{\sum_{k=1}^n DE_k}{TLCR}$$

Da una idea de la cantidad de defectos encontrados en el proyecto, por lo tanto sirve para analizar la efectividad del proceso de revisión y además permite analizar el mejoramiento continuo de la calidad del proceso de desarrollo, pues los proyectos deben tender a tener cero defectos y a esto deben contribuir las Revisiones. Se puede utilizar un modelo similar al utilizado para mostrar la métrica de densidad de defectos.

#### 4. CONCLUSIONES

El Proceso de Revisiones tiene vital importancia en el Aseguramiento de Calidad en una empresa que desarrolla software. Su planificación, control y seguimiento incide directamente en la calidad del producto final. Por tanto, establecer un adecuado Sistema de Aseguramiento de Calidad en la empresa cubana es de suma importancia en las condiciones actuales del país, en las que se quiere potenciar la industria de producción de software.

En este trabajo se ha expuesto una propuesta de procedimiento que hace uso de las mejores prácticas utilizadas en la industria internacional de software, adecuándolas a las condiciones de la PYME, aplicable a la industria nacional. La introducción del procedimiento permitirá a la empresa mejorar la calidad de sus productos de software y de esta forma crear una cultura en el uso de estándares de calidad, que comienza por la disciplina de los individuos involucrados en el proceso tecnológico.

La efectividad del proceso puede ser medida en su planificación, control y seguimiento y mejoramiento continuo, a partir del registro de las diferentes métricas propuestas en este trabajo. Su análisis dotará a la empresa de elementos que permiten detectar y mejorar el desarrollo de las Revisiones y el propio Proceso de Desarrollo de Software.

#### REFERENCIAS

- ÁLVAREZ, S. (2000): "Aplicación del modelo CMM a la empresa Segurmática", Instituto Superior Politécnico "José Antonio Echeverría", La Habana.
- DELGADO, M. (2003): "Inspecciones a Proyectos de Software, Garantía de Calidad", **Memorias de Informática 2003**, Marzo, La Habana, Cuba.
- \_\_\_\_\_ (2003): "Revisiones a proyectos de Software, una tarea que requiere experiencia", **Memorias de la Conferencia Internacional Gerencia y Tecnología**, Abril, La Habana, Cuba.
- FEBLES, A. (2001): "CASE Corporativo para el proceso de control de cambios". Tesis presentada en opción al título de Master en Informática Aplicada. Instituto Superior Politécnico "José Antonio Echeverría".
- \_\_\_\_\_ (2004): "Un modelo de referencia para la gestión de configuración en la pequeña y mediana empresa de software". Tesis presentada en opción del grado de Doctor en Ciencias Técnicas. Instituto Superior "José Antonio Echeverría"..
- FOX, Ch. and W. FRAKES (1997): "Elements of the Quality Paradigm", **Communications of ACM**, Junio, 40(6).
- HUMPHREY, W. (1997): "Managing technical people: innovation, teamwork, and the software process", Addison Wesley Longman.
- JACOBSON, I. (2000): "El Proceso Unificado de Desarrollo de Software", Addison Wesley Longman Inc.
- KAN, S. (1995): "Metrics and Models in Software Quality Engineering", Addison Wesley Longman, Inc.
- MARKUS, M. (2000): "¿Fallidos proyectos de software?, Ya no más", **Quality Progress**, 11(33), 116-117.
- MORENO, B. (2003): "Discurso en el Primer Taller de Calidad de Software", **Informática 2003**, La Habana, Cuba,

- OKTABA, H. (2002): "Procesos de Desarrollo de Software", **Primer Taller Internacional de Calidad en desarrollo de software**, México.
- \_\_\_\_\_ (2003): "El modelo de madurez de capacidades mexicano", **Conferencia pronunciada en el evento Calidad 2003 en la Convención Informática 2003**, Cuba.
- PAUL, M. (1999): "Using the Software CMM With Good Judgment", **ASQ Software Quality Professional**, 1, 19-29.
- QUINTANILLA, G. (2002): "Modelos y Estándares de Referencia para la Gestión de Calidad del Software", **Primer Taller Internacional de Calidad en desarrollo de software**, México.
- Rational Unified Process (2001): Rational Software Corporation. "Rational Unified Process", Version 2001A.04.00, Copyright 1987-2001.
- REO, D. (2002): "La gallina, el cerdo y el modelo CMM", **América XXI**, sep. 4, Disponible en: <http://www.americaxxi.cl/modules.htm>.
- REYNOLDS, G. (1995): "Information Systems for Managers", third edition, West Publishing Company.
- SCHULMEYER, G. (1997): "Handbook of Software Quality Assurance", Prentice.
- SEI (2000): "Software Engineering Measurement and Analysis Team", Process Maturity Profile of the software community 2000 mid-year update.
- SEI (2001): **Interactive Process Maturity Profile of the software Community**, SEMA.