Memoria del 6to. Congreso Internacional de Investigación
en Ciencias Computacionales CIICC'99
22 a 24 de Septiembre de 1999, Cancún, México

Propuesta de un Marco Formal para la Auditoría del Proceso de Mantenimiento del Software

A Proposal of Framework for Auditing the Software Maintenance Process

Francisco Ruiz, Mario Piattini, Macario Polo, Coral Calero
Grupo Alarcos - Dep. de Informática
{fruiz, mpiattin, mpolo, ccalero}@inf-cr.uclm.es
fax: 34-926-295354; tlf/phone: 34-926-295300
Escuela Superior de Informática – Universidad de Castilla-La Mancha
Ronda de Calatrava, 7 – 13071-Ciudad Real (España / Spain)

Resumen:

Una de las principales causas de la llamada "crisis del software" ha sido la poca importancia que se le ha dado al proceso de mantenimiento desde todos los colectivos afectados (gestores de empresas, responsables de centros de proceso de datos, informáticos, usuarios y auditores). En este documento, presentamos una propuesta de un marco formal para la auditoría del proceso de mantenimiento del software basada en los estándares oficiales (ISO 12207, ISO 14764) y en la metodología CobiT para la auditoría de sistemas de información propuesta por la ISACF (Information Systems Audit and Control Foundation). El trabajo se enmarca dentro de los proyectos MANTEMA [1] y MANTICA cuyo objetivo general es construir un marco metodológico y unas herramientas para abordar el mantenimiento del software de forma general e integrada.

Abstract:

One of the main causes of the "crisis of software" has been the little importance that all affected communities (managers, responsible for centers of data process, computer engineers, users and auditors) have given to the maintenance process. In this document, we present a proposal of a framework for the audit of the software maintenance process based on the official standards (ISO 12207, ISO 14764) and the methodology CobiT for auditing of information systems proposed by the ISACF (Information Systems Audit and Control Foundation). The work is include inside the projects MANTEMA and MANTICA whose general objective is to build a methodological framework and some tools to approach the software maintenance in a general and integrated way.

Palabras clave / Keywords:

Mantenimiento del Software, Auditoría de Sistemas de Información, CobiT
Software Maintenance, Auditing Information Systems, CobiT

1. Introducción

Múltiples estudios señalan que el mantenimiento es la parte más costosa del ciclo de vida del software. Estadísticamente está comprobado que el coste de mantenimiento de un producto software a lo largo de toda su vida útil supone más del doble que los costes de su desarrollo. La

[1] El proyecto MANTEMA está financiado por la empresa Atos ODS y por el Ministerio de Industria y Energía de España (iniciativa ATYCA). El proyecto MANTICA está financiado por la Unión Europea (CICYT 1FD-097).

tendencia es creciente con el paso del tiempo y, en general, el porcentaje de recursos necesarios para mantenimiento se incrementa a medida que se produce más software [3].

Una causa directa de los grandes costes del mantenimiento del software (MS) es que el coste relativo aproximado de reparar un defecto aumenta considerablemente en las últimas etapas del ciclo de vida del software [9], de forma que la relación entre el coste de detectar y reparar un defecto en la fase de análisis de requisitos y en la fase de mantenimiento es de 1 a 100 respectivamente (ver figura 1).

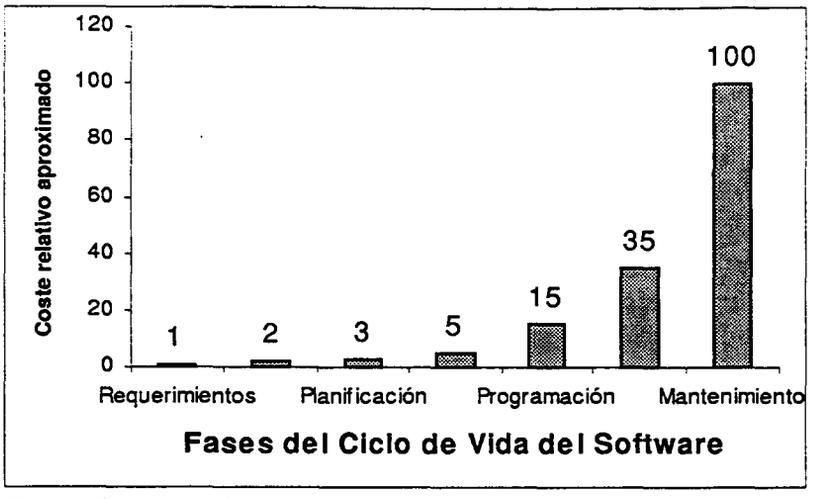

Figura 1. Coste relativo aproximado de detectar y corregir defectos

Cuando se planifican los costes de mantenimiento, los analistas-programadores experimentados tienen la impresión de que parece como si el MS fuese un *iceberg* del cual sólo se percibe una pequeña parte, pero bajo cuya superficie se esconde una gran cantidad de problemas potenciales y de costes encubiertos. En la parte sumergida de este iceberg se ocultan otros costes, menos tangibles que los monetarios, pero que pueden ser causa de muchas preocupaciones.

En suma, un coste final del MS es la reducción que se produce en la productividad de los informáticos al iniciar el mantenimiento de aplicaciones antiguas. Algunos autores [11] han calculado reducciones de la productividad -medida en líneas de código (LDC) por persona y mes- de 40 a 1, es decir, el coste de mantener una línea de código puede llegar a ser 40 veces más alto que en el proceso de desarrollo.

A la vista de la importancia del MS (en términos económicos y de recursos consumidos), parece necesario que se tenga especialmente en cuenta al realizar auditoría de sistemas de información (ASI) y, especialmente, cuando se trata de auditar los procesos para producir y poner en producción los productos software. Frente a esta evidencia, la realidad es que hasta ahora el MS no ha sido tenido en cuenta en los procedimientos y normas establecidos para la ASI.

Entre otras posibles causas de esta situación, creemos que se encuentra el hecho de que es muy reciente la atención prestada al MS desde el mundo de la ingeniería del software, prueba de ello es que los estándares internacionales para el proceso de mantenimiento del software tienen muy pocos años [4] o acaban de publicarse [7] y que casi no existen metodologías para abordar

las particularidades que dicho proceso tiene respecto del proceso de desarrollo de software [12].

En este trabajo presentamos una propuesta para la Auditoría del proceso de mantenimiento del software (APMS) que toma como punto de partida la arquitectura de procesos del ciclo de vida del software definida en el estándar ISO 12207 [6]. En este estándar el MS y la auditoría son dos procesos definidos. El MS es uno de los cinco procesos principales (junto con la adquisición, el suministro, el desarrollo y la explotación), mientras que la auditoría es uno de los ocho procesos de soporte (ver figura 2).

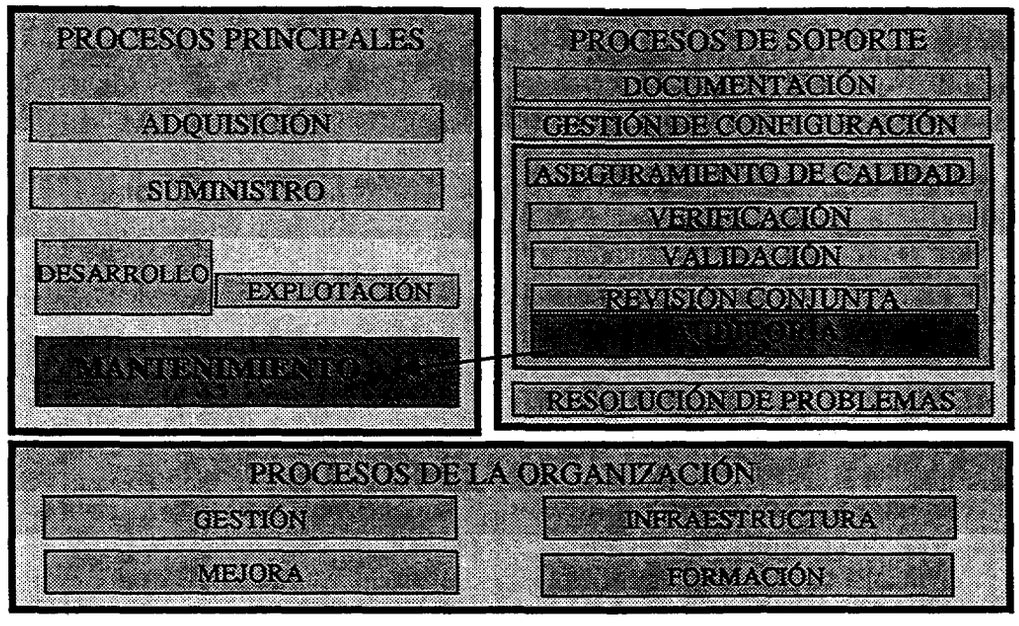

Figura 2. Estructura de procesos en ISO 12207

Nosotros nos centramos en el segundo (la auditoría) como soporte o herramienta de control para el primero (el MS). Para ello, comenzamos realizando una presentación de los conceptos y marcos utilizados (apartados 2 y 3):

- para el proceso de mantenimiento del software: el estándar ISO 14764 [7], y
- para la auditoría de sistemas de información: la metodología CobiT [5].

Después, presentamos la propuesta de adaptación de la metodología CobiT para cumplir con la norma ISO 14764 (apartado 4). Por último, acabamos realizando unas conclusiones y la exposición de trabajos actuales y futuros (apartado 5).

2. El Proceso de Mantenimiento del Software (PMS)

El PMS incluye las diversas actividades y tareas cuyo objetivo es modificar un producto software existente y ya puesto en explotación preservando su integridad. Este proceso es activado cuando el producto software sufre modificaciones en el código o en la documentación asociada con el objetivo de:

- localizar y eliminar defectos, normalmente detectados por un funcionamiento incorrecto

(mantenimiento correctivo);
- adaptar el software a cambios en el entorno operativo (hardware y/o software) (mantenimiento adaptativo);
- mejorar o añadir nuevas funcionalidades requeridas por los usuarios (mantenimiento perfectivo); o
- mejorar las propiedades del software (calidad, mantenibilidad, etc.) sin alterar las especificaciones funcionales (mantenimiento preventivo).

El estándar ISO 14764 establece cuatro tipos de mantenimiento que coinciden con los cuatro objetivos anteriores. En algunas metodologías se amplían y precisan estos tipos de mantenimiento [13]. También se establecen diversos aspectos del PMS que deberemos tener en cuenta al planificar la auditoría de dicho proceso:

- Existencia de contratos o acuerdos de mantenimiento entre la organización que realiza el mantenimiento (mantenedor) y la propietaria del software (cliente). Algunas veces entrará en juego un tercero cuando el software es utilizado por una organización diferente del cliente (el usuario).
- Disponibilidad de herramientas para el mantenimiento (un tipo especial de herramientas CASE).
- Técnicas de medida del software para evaluar sus propiedades de calidad.
- Documentación de todo el proceso.
- Los estudios realizados indican que los costes del mantenimiento pueden verse reducidos significativamente si durante el proceso de desarrollo del software se incluye como objetivo su mantenibilidad. Para ello es importante que, si es posible, el futuro mantenedor participe también durante la fase de desarrollo.
- Existencia de un procedimiento organizado y controlado para pasar desde la fase de desarrollo a la de mantenimiento (transición de la responsabilidad sobre el producto software desde el desarrollador al mantenedor).

Ya hemos comentado la importancia del mantenimiento dentro del ciclo de vida de un producto software. Por ello, la norma ISO propone la necesidad de una planificación estratégica del PMS, cuyo objetivo es preparar los recursos (humanos, materiales y de gestión) para poder realizar el citado proceso de mantenimiento con garantías de éxito. Para ello, se definen las actividades de MS que deberán ser realizadas. Desarrollar la estrategia del MS consta de los siguientes elementos:

1) Describir el proceso: establecer el alcance, particularizar el proceso a partir de las normas establecidas, designar quién será el mantenedor, y realizar una estimación de los costes.
2) Definir las actividades de mantenimiento y organizativas necesarias.
3) Realizar un análisis de los recursos necesarios (personal, software, hardware, financieros, instalaciones, documentación, datos, etc.).

Una de las principales actividades organizativas es la elaboración del plan de mantenimiento; un documento donde se especifican todos los aspectos a considerar: porqué es necesario el mantenimiento, quién hará qué trabajo, cuáles son los papeles y responsabilidades de cada uno, cómo será realizado el trabajo, qué recursos estarán disponibles, donde se realizará el trabajo, y cuando se realizará.

El PMS propiamente dicho consta de las actividades y tareas necesarias para modificar un producto software existente preservando su integridad (ver figura 3). Dichas actividades y tareas son responsabilidad del mantenedor.

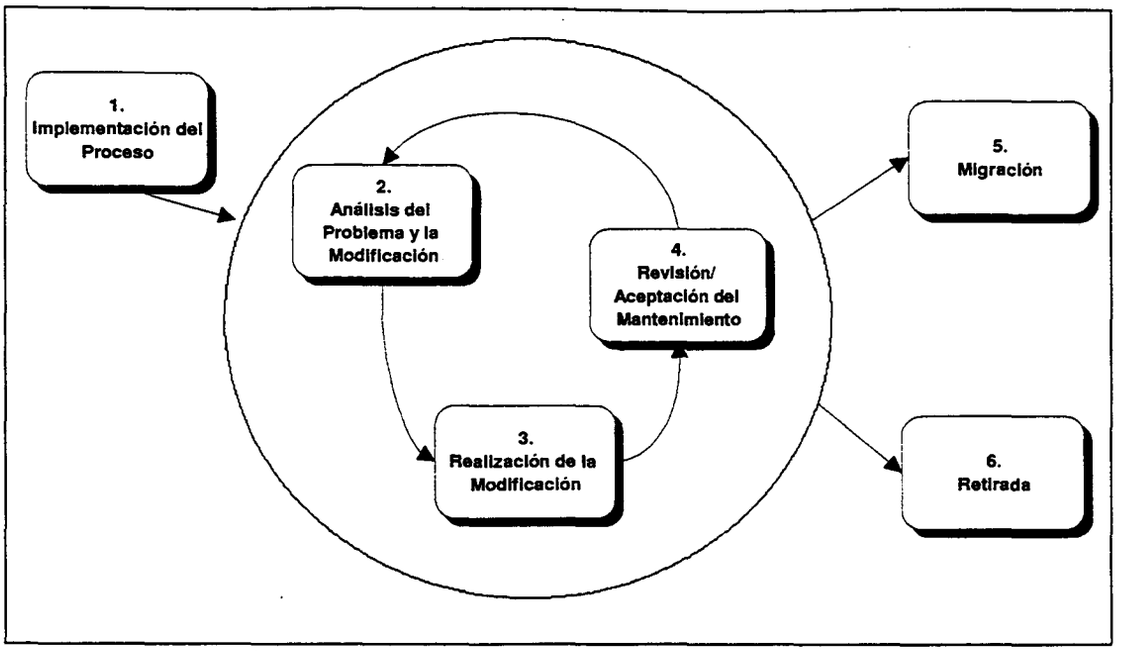

Figura 3. Actividades del Proceso de Mantenimiento del Software

3. La metodología CobiT para Auditoría de Sistemas de Información (ASI)

No es extraño leer en los medios de comunicación informaciones sobre empresas que, habiendo sido objeto de auditoría y obtenido un informe favorable, poco después se detecten grandes problemas de control, agujeros financieros, fraudes, etc. Distintos informes internacionales lo achacan a la revisión parcial que se hace del sistema de control interno. Se hace necesario pues, un enfoque que, a la hora de realizar la auditoría, considere el sistema de información globalmente; es decir, que tenga en cuenta de manera conjunta, los procesos manuales y los informáticos. El sistema de información de la empresa es uno, aunque ciertos procesos se realicen de forma manual y otros mediante la informática. El auditor utiliza, en cada caso, las herramientas y los procedimientos más adecuados en función de la realización manual o informática de las actividades.

La propuesta CobiT [5] supone un paso, seguramente el más importante, en dicho camino. La filosofía de CobiT asimila los principios de reingeniería de empresas, y divide las funciones que ha de realizar un sistema de información en procesos que, a su vez, están subdivididos en actividades y tareas más simples. Los sistemas de información están orientados a los procesos y por tanto su auditoría se debe adaptar a estos conceptos.

La estructura (*framework*) de CobiT comienza a partir de una premisa simple y pragmática: *Los recursos de las Tecnologías de la Información y las Comunicaciones (TIC) se han de gestionar mediante un conjunto de procesos agrupados de forma natural para que proporcionen la información que la empresa necesita para alcanzar sus objetivos.*

Para ello, se definen 34 objetivos de control generales (OCGs) [2], uno para cada uno de los procesos de las TIC. Estos procesos están agrupados en cuatro grandes dominios: planificación y organización, adquisición e implantación, suministro y soporte, y supervisión. Esta estructura cubre todos los aspectos de la información y de las tecnologías que le sirven de soporte [8].

Además, en la estructura de CobiT se destacan los efectos de los recursos en TIC (datos, aplicaciones, tecnología, instalaciones y personal) junto con los requisitos o criterios que debe satisfacer la información:

- Requisitos de calidad: calidad, coste, suministro;
- Requisitos fiduciarios [2]: efectividad y eficiencia de las operaciones, fiabilidad de la información, legalidad (cumplimiento de las leyes y normas);
- Requisitos de seguridad: confidencialidad, integridad, disponibilidad.

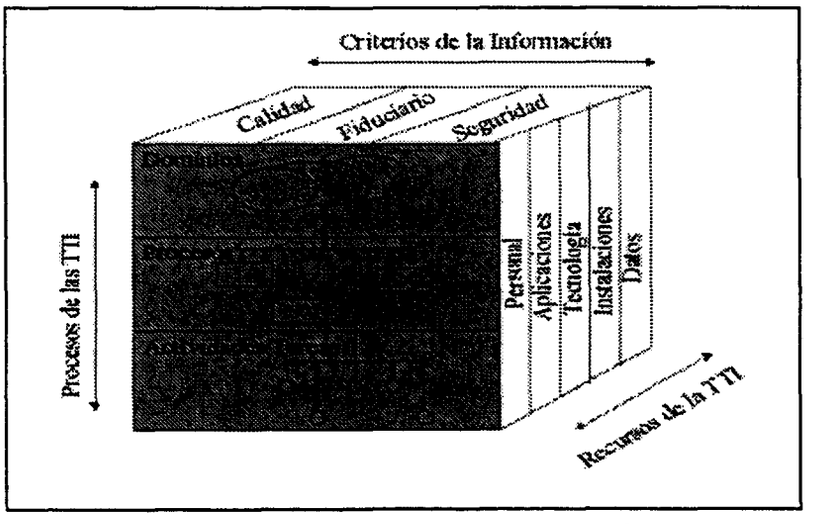

Figura 4. Las tres dimensiones conceptuales de CobiT

En suma, la estructura conceptual se puede enfocar desde tres puntos de vista (ver figura 4):

1) Los recursos de las TIC,
2) Los criterios empresariales de la información, y
3) los procesos de las TIC.

---

[2] Un control se define como "*las normas, estándares, procedimientos, usos y costumbres y las estructuras organizativas, diseñadas para proporcionar garantía razonable de que los objetivos empresariales se alcanzarán y que los eventos no deseados se preverán o se detectarán, y corregirán*".
Un objetivo de control se define como "*la declaración del resultado deseado o propuesto que se ha de alcanzar mediante la aplicación de procedimientos de control en cualquier actividad de las TIC*".

Estas diferentes vistas permiten que se pueda acceder a la estructura de manera eficiente desde la óptica de interés de cada implicado: directivo de empresa, gestor de TIC, responsable de procesos, técnico en TIC o usuario.

4. Adaptación de CobiT al Proceso de Mantenimiento del Software

Las fuentes bibliográficas más recientes dedicadas a la auditoría de sistemas de información [1], [10] dedican muy poca o ninguna atención a la auditoría del MS. Por dicha causa, hemos estado trabajando en una propuesta original que presentamos a continuación.

En CobiT, los 34 OCGs propuestos se concretan en 302 objetivos de control detallados (OCDs). En la metodología MANTEMA [12] hemos utilizado como punto de partida para la auditoría del PMS los siguientes objetivos de control (seleccionados y extraídos entre los 34 generales y los 302 detallados):

Dominio / Objetivos Generales / Objetivos Detallados:

AI - Adquisición e Implantación
- AI01 – Identificación de soluciones
  - 1.15 Mantenimiento del software por terceros
- AI02 – Adquisición y mantenimiento de aplicaciones software
  - 2.2 Cambios grandes en sistemas existentes
- AI05 – Instalación y acreditación de sistemas
  - 5.3 Conversión
- AI06 – Gestión de cambios
  - 6.1 Iniciación y control de los requerimientos de cambio
  - 6.2 Valorar impacto
  - 6.3 Definir el control de cambios
  - 6.4 Actualización de documentación y procedimientos
  - 6.5 Autorización del mantenimiento
  - 6.6 Política de versiones del software
  - 6.7 Distribución del software

DS - Suministro y Soporte
- DS09 – Gestión de la configuración
  - 9.1 Registrar la configuración
  - 9.2 Configuración básica
  - 9.3 Contabilizar los estados pasados
  - 9.4 Control de la configuración
  - 9.6 Almacenar el software

Además de los anteriores, existen otros objetivos generales (OCGs) y detallados (OCDs) que se relacionan fundamentalmente con el proceso de desarrollo de software, pero que también son de aplicación al mantenimiento debido a que durante la actividad de realización de la modificación el mantenedor tiene que realizar algunas de las tareas típicas del desarrollo del

software (análisis, diseño, codificación, prueba, ...).

No todos los objetivos incluidos en la lista anterior tienen la misma importancia dentro del PMS (según se define en el estándar ISO 14764). El dominio en el que se incluyen la mayoría de las actividades del PMS es el de "Adquisición e Implantación". Dentro de este dominio, el OCD AI01.15 (Mantenimiento del software por terceros) pertenece realmente - a pesar del nombre- al proceso de adquisición, en este caso adquiriendo (contratando) el servicio de mantenimiento mediante externalización u *'outsourcing'*. El análisis de los objetivos detallados del OCG AI02 (Adquisición y mantenimiento de aplicaciones software) permite comprobar que, también a pesar de incluir la palabra mantenimiento en el nombre, no se corresponde realmente con el PMS salvo en el OCD AI02.2 (Cambios grandes en sistemas existentes) que se refiere a situaciones que requieren mucho mantenimiento adaptativo. El OCD AI05.3 (Conversión) está relacionado con la actividad de migración dentro del PMS.

En realidad, dentro del dominio de "Adquisición e Implantación", el OCG que realmente está asociado al PMS es el AI06 (Gestión de cambios). Todos los OCDs que lo integran están directamente asociados con las actividades del PMS (ver figura 3). En la tabla 1 mostramos los siete OCDs de la Gestión de cambios e indicamos las actividades del PMS relacionadas. No aparece la actividad de migración porque dicha actividad se produce sólo en el caso de mantenimiento adaptativo (OCD AI02.2 ya comentado). Tampoco aparece la actividad de retirada porque no se puede considerar directamente relacionada con la gestión de cambios.

Objetivos de Control Detallados (CobiT)	**Actividades PMS relacionadas**
6.1 Iniciación y control de los requerimientos de cambio	Implementar el Proceso
6.2 Valorar impacto	Análisis del Problema y Modificación Realización de la Modificación
6.3 Definir el control de cambios	Implementar el Proceso
6.4 Actualización de documentación y procedimientos	Realización de la Modificación
6.5 Autorización del mantenimiento	Revisión/Aceptación del mantenimiento
6.6 Política de versiones del software	Implementar el Proceso
6.7 Distribución del software	Realización de la Modificación

Tabla 1. Objetivos de Control de la Gestión de Cambios vs Actividades del PMS.

En el dominio de Suministro y Soporte, el OCG DS09 (Gestión de la configuración) está directamente relacionado con el PMS, pero se corresponde con el proceso del mismo nombre definido en la norma ISO 12207 como uno de los procesos de soporte. Por tanto, no lo tendremos en cuenta en nuestra propuesta.

Todas estas disfunciones se deben, fundamentalmente, al diferente modelo de procesos utilizado por CobiT y por los estándares ISO 12207 y 14764. Por esta razón, en la metodología MANTEMA 2.0 para la Gestión Integral del Mantenimiento del Software, para poder utilizar la propuesta CobiT de manera coherente con el PMS propuesto por ISO proponemos modificar la lista de OCGs sustituyendo el AI06 'Gestión de cambios' por 'Gestión del proceso de mantenimiento del software', en el cuál incluimos también los OCDs AI02.2 (Cambios grandes en sistemas existentes) y AI05.3 (Conversión) por las razones ya comentadas. Además, los OCDs del OCG AI06 se reestructuran en función de las actividades y tareas del PMS en ISO 14764.

La gestión del PMS pasa a ser un objetivo de control general dentro del dominio de 'Adquisición e Implantación' ya que el MS es un proceso básico para la correcta implantación (explotación) de un sistema de información.

A continuación se resume el resultado de la primera versión que hemos realizado de objetivos de control detallados:

Dominio: Adquisición e Implantación

Objetivo General: AI06 - Gestión del proceso de mantenimiento del software

Descripción: *las actividades del negocio se realizan sin interrupciones imprevistas y el software de los sistemas de información existentes se adapta a las nuevas necesidades.*

Objetivos de Control Detallados:

6.1 Cambios en el entorno operativo: existe un procedimiento organizado para realizar la migración de un producto software desde un entorno operativo antiguo a otro nuevo.

6.2 Retirada del software: la metodología de desarrollo y/o mantenimiento de software incluye un procedimiento formal para la retirada de un producto software cuando ha concluido su ciclo de vida útil.

6.3 Tipos de mantenimiento: están categorizados los tipos de mantenimiento del software y para cada tipo se han planificado las actividades y tareas a realizar.

6.4 Acuerdo de mantenimiento: las relaciones entre el mantenedor [3] y el cliente y las obligaciones de cada uno están establecidas en un acuerdo o contrato de mantenimiento.

6.5 Mejora de la calidad del proceso: la metodología empleada para el mantenimiento del software incluye técnicas para aumentar la mantenibilidad (facilidad de mantenimiento).

6.6 Planificación del mantenimiento: Existe un plan de mantenimiento que incluye el alcance del mantenimiento, quién lo realizará, una estimación de los costes y un análisis de los recursos necesarios.

6.7 Procedimientos para solicitudes de modificación (SM): existen procedimientos normalizados para iniciar, recibir y registrar SMs.

6.8 Gestión y control de cambios: el mantenedor tiene establecido un interface organizacional para que el proceso de mantenimiento pueda verse beneficiado por el proceso de gestión de la configuración.

6.9 Análisis y valoración de las SMs: las SMs son categorizadas y priorizadas, y existen mecanismos bien estructurados para evaluar su impacto, costes y criticidad.

6.10 Verificación de los problemas: el mantenedor replica o verifica que realmente existe el problema que originó la SM.

6.11 Registro de las SMs: el mantenedor documenta y registra las SMs, con sus análisis, valoraciones y verificaciones.

6.12 Aprobación: dependiendo del tipo de mantenimiento de una SM, existen procedimientos formales que detallan el tipo de aprobación que el mantenedor debe obtener antes y después de realizar la modificación.

---

[3] El mantenedor y el cliente (propietario o usuario del software mantenido) pueden pertenecer a la misma organización.

6.13 Realización de las modificaciones: para realizar las modificaciones, el mantenedor utiliza la misma metodología establecida para el proceso de desarrollo del software adaptada al proceso de mantenimiento.

6.14 Actualización de la documentación: la documentación (informes técnicos, manuales, etc.) afectada por una SM es actualizada después de realizada la modificación.

5. Conclusiones y trabajos pendientes

El mantenimiento es la fase más costosa de todo el ciclo de vida del software. Por esta razón, es importante que desde la Ingeniería del Software se le dedique la atención que merece. Con este objetivo, presentamos una propuesta para abordar la auditoría del proceso de mantenimiento del software basada en:

- el estándar ISO 14764 para el proceso de mantenimiento del software, y
- la metodología CobiT para la auditoría de sistemas de información.

En dicha propuesta hemos realizado un análisis de todos los objetivos de control incluidos en CobiT y hemos seleccionado los que tienen relación con el proceso de mantenimiento del software. La lista resultante la hemos cambiado definiendo un objetivo de control general llamado 'Gestión del proceso de mantenimiento del software'. Este objetivo general lo hemos precisado en 14 objetivos de control detallados que amplían considerablemente los incluidos en CobiT.

Con esta propuesta establecemos un marco formal (en el cual la auditoría es uno de los procesos de soporte al proceso de mantenimiento) que hemos incluido como parte de la versión 2 de la metodología MANTEMA para la gestión integral del mantenimiento del software.

Los resultados obtenidos están siendo validados en entornos reales de mantenimiento de grandes proyectos software mediante la colaboración de la empresa Atos ODS, una de las principales compañías europeas en el campo de la externalización y '*outsourcing*' de servicios informáticos. Los comentarios y sugerencias obtenidos serán utilizados para reformar la lista de objetivos de control, para cambiar sus descripciones y para elaborar una colección de técnicas de control útiles para detectar si se satisface cada objetivo de control.

6. Referencias

[1] Champlain, J., Auditing Information Systems. A Comprehensive Reference Guide. John Wiley & Sons. USA, 1998.

[2] Commitee of Sponsoring Organizations oh the Treadway Commison. *Internal Control - Integrated Framework.* American Institute of Certified Accountants. New Jersey, USA 1994.

[3] Hanna, M., Maintenance "Burden Begging for a Remedy". *Datamation,* abril 1993, pp. 53-63.

[4] IEEE, std 1219: *Standard for Software Maintenance.* IEEE Computer Society Press. USA, 1993.

[5] ISACF, *CobiT: Governance, Control and Audit for Information and Related Technology*, 2nd edition. Information Systems Audit and Control Foundation. USA, 1998.

[6] ISO/IEC 12207: *Information Technology – Software life cycle processes*. ISO/IEC JTC1/SC7 Secretariat. Canadá, 1995.

[7] ISO/IEC 14764: *Software Engineering – Software Maintenance*. ISO/IEC JTC1/SC7 Secretariat. Canadá, 1998.

[8] Peña, E., Objetivos de Control y Estructura de CobiT. JAI'98, *I Jornadas de Auditoría Informática*. Grupo Alarcos (editores). Ciudad Real, España 1998.

[9] Piattini, M. G., Ruiz, F., Polo, M., Villalba J., Fernández, I., Bastanchury, T. y Martínez, M.A., *Mantenimiento del Software. Conceptos, métodos, herramientas y outsourcing*. Ed. Ra-Ma. Madrid, España 1998.

[10] Piattini, M., del Peso, E., *Auditoría Informática. Un enfoque práctico*. Ed. Ra-Ma. Madrid, España 1998.

[11] Pigoski, T. M., *Practical Software Maintenance. Best Practices for Managing Your Investment*. Ed. John Wiley & Sons. USA, 1996.

[12] Polo, M., Piattini, M., Ruiz, F., Calero, C. MANTEMA: A Complete Rigorous Methodology for Supporting Maintenance based on the ISO/IEC 12207 Standard. CSMR'99, *Third European Conference on Software Maintenance and Reengineering*. IEEE Computer Society Press. Amsterdam, Holanda 1999.

[13] Ruiz, F., Piattini, M., Polo, M., Calero, C. Maintenance Types in the MANTEMA Methodology. ICEIS'99, *First International Conference on Enterprise Information Systems*. Setúbal, Portugal 1999.