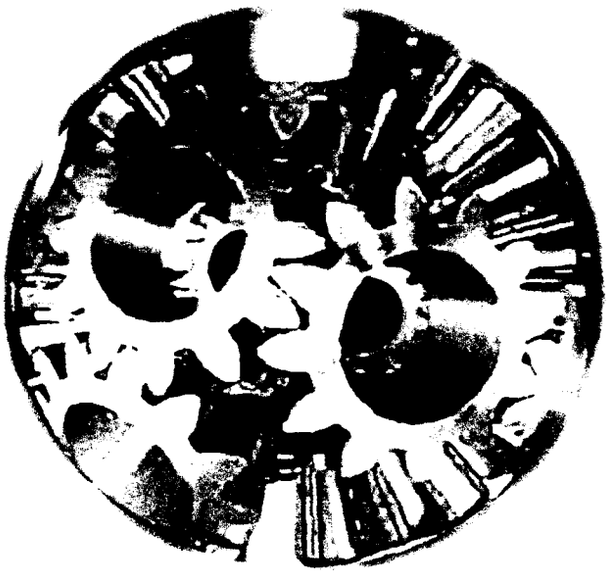
Auditoría y Control de Sistemas de Información para



CASI'99

II Congreso Nacional de
AUDITORIA y CONTROL
de Sistemas de Información

Edita: SERVICIO DE PUBLICACIONES
Camino de Vera, s/n
46071 VALENCIA
Tel. 96-387 70 12
Fax. 96-387 79 12
Ref. 2017

Imprime: REPROVAL, S.L.
Tel. 96-369 22 72

Depósito Legal: V-4021-1999
I.S.B.N.: 84-7721-808-0

CASI'99

Organizan:

Universidad Politécnica de Valencia.
Departamento de Organización de Empresas.
Facultad de Informática.
Escuela Universitaria de Informática.

Patrocinan:

Asociación de Doctores, Licenciados e Ingenieros en Informática.
Consejo General de Colegios de Economistas de España.
Organización de Auditoría Informática.
Information Systems Audit and Control Association.

Empresas Patrocinadoras:

ACL
Atos ODS
Microsoft
SAP

INDICE

PONENCIAS

PREFACIO

Dr. Mario Piattini Velthuis, CISA

El presente volumen contiene los trabajos presentados en el II Congreso Nacional de Auditoría y Control de Sistemas de Información (CASI'99), celebrado en Valencia durante los días 21, 22 y 23 de octubre de 1999.

Esta segunda edición supone la consolidación de este foro de encuentro, celebrado en su edición anterior en Ciudad Real, que nació con la intención de que los expertos e investigadores en el área de la Auditoría de los Sistemas de Información puedan compartir sus experiencias, divulgar resultados y dar a conocer sus trabajos.Este II Congreso se organizó en cuatro conferencias invitadas, un panel de expertos, cuatro seminarios técnicos, una mesa redonda, cuatro presentaciones de empresas y cuatro sesiones técnicas.

Para las sesiones técnicas se recibieron 26 trabajos de los cuales se aceptaron 18. El nivel y la calidad de todas las contribuciones enviadas dificultó la siempre ardua labor del Comité de Programa. Cada trabajo fue evaluado por los miembros del comité de programa, y las sugerencias de éstos incorporadas en las versiones finales que se presentan en este volumen.

Quería agradecer el esfuerzo desarrollado por los miembros del Comité de Programa en todo el proceso de revisión de los trabajos. Las conferencias, así como el panel de expertos, los seminarios, la presentación de empresas y la mesa redonda fueron realizadas por invitación.

Quisiera dejar constancia desde aquí a todos los conferenciantes y ponentes en estos eventos, por su extraordinaria colaboración. Este congreso no hubiera podido realizarse sin el trabajo de todo el comité organizador, a quien quería agradecer su generosidad por las muchas horas de trabajo entregadas; en especial al director del mismo: Rafael Bernal, y a Inmaculada Vilar de la Secretaría Técnica.

Agradecimiento que ha de hacerse extensivo a Dª Marina Touriño, presidenta de la OAI (Organización de Auditoría Informática) por la enorme cantidad de horas dedicadas para lograr que este congreso reúna los principales expertos de auditoría informática del país.

Ciudad Real, 17 de septiembre de 1999
MARIO PIATTINI VELTHUIS
Presidente del Comité de Programa

COMITÉ ORGANIZADOR

F. Barber Sanchís. *Decano de la Facultad de Informática U.P.V.*

A. González del Río. *Subdirector de Relaciones Externas E.U.I.*

R. Mª Bernal Montañés, *CISA. ALI, UPV*

E. Peña Ramos. *Univ. Málaga*

I. Gil Pechuán. *ViceDecano de Relaciones Externas F.I. U.P.V.*

F. Ruiz González. *Director Escuela Superior de Informática UCLM*

F. J. García Martínez. *DSIC-UPV.*

COMITÉ DE PROGRAMA

F. Barber Sanchis., *ATI, UPV. Decano de la Fac.de Informática.*
V. Carrascosa. *UNED*
O.Coltell Simon. *Univ. Jaime I*
M. A. Davara. *Inst. Inf. Jurídica*
J. de la Peña. *Auditor*
E. del Peso, *ALI, ATI, IEE*
J. del Valle Fernández,*CISA. ALI, ATI, Caja Rural Valencia*
C. Fernández,CISA . *ALI, U. P Salamanca, Microsoft, BSA*
M. Ferrando. *Vicerector UPV*
J.M.Ferrer, *ALI*
F. Fons. *ATI, BANCAJA*
F.Forero, Presidente *ATI Valencia*
F. J. García Martínez. *ALI, UPV*
I. Gil Pechuán., *UPV ViceDecano Fac. De Informática.*
J. Gutierrez. *Univ. del País Vasco*
J. Antonio Rodero – *AII*
A. Jiménez. *GeneralItat Valenciana.*
A.Juarros, *ALI, OAI*
V. Izquierdo. *MINER*
J. Páez. *CSIC*
F.Peris Bonet, *CEU, Universidad de Valencia*
M. Piattini Velthuis, CISA/*UCLM, ATI, ALI (Presidente)*
M. Angel Ramos, CISA *ALI, IEE*
A. Ribagorda Garnacho. *Univ.Carlos III de Madrid*
Jesús Rivero - *Presidente Fundación DINTEL*
F. Sanchis. *UPM*
M. Toro. *Univ.de Sevilla*
F. Villarubia,CISA *ALI, Caja Castilla-La Mancha*

AUDITORÍA DEL MANTENIMIENTO DEL SOFTWARE:

PROPUESTA DE OBJETIVOS DE CONTROL

Francisco Ruiz;
Mario Piattini;
Macario Polo;
Coral Calero;

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). Por contra, los costes de mantenimiento tienen un peso muy importante en los costes totales de un producto software a lo largo de su ciclo de vida completo. Por ello, se hace necesario que desde el campo de la Auditoría de Sistemas de Información (ASI) se le dedique a esta fase del ciclo de vida la atención que se merece. En este trabajo presentamos una propuesta de un marco formal para la auditoría del proceso de mantenimiento del software (PMS) y proponemos una lista de 14 objetivos de control. La propuesta está basada en los estándares oficiales (ISO 12207, ISO 14764) y en la metodología CobiT para la auditoria de sistemas de información elaborada por la ISACF (Information Systems Audit and Control Foundation). Este trabajo se enmarca dentro de los proyectos MANTEMA y MANTICA [16] cuyo objetivo general es construir un marco metodológico y unas herramientas para abordar el mantenimiento del software de forma general e integrada.

Palabras clave:

Mantenimiento del Software, Auditoría del Mantenimiento, Objetivos de Control, CobiT.

[16] 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).

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. En el apartado 2 se presenta un análisis de las causas que originan los altos costes del mantenimiento del software (MS).

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 [IEEE, 1993] o acaban de publicarse [ISO/IEC, 1998] y que casi no existen metodologías para abordar las particularidades que dicho proceso tiene respecto del proceso de desarrollo de software [Polo et al, 1999].

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 [ISO/IEC, 1995]. En este estándar el MS y la auditoría son dos procesos expresamente definidos. En la propuesta presentada, nos centramos en el segundo (la auditoría) como soporte o herramienta de control para el primero (el MS). Para ello, en los apartados 3 y 4 se presentan los conceptos y marcos utilizados:

- para el proceso de mantenimiento del software: el estándar ISO 14764 [ISO/IEC, 1998], y
- para la auditoría de sistemas de información: la metodología CobiT [ISACF, 1998].

En el apartado 5 se presenta la propuesta de adaptación de la metodología CobiT para cumplir con la norma ISO 14764 y por último, en el apartado 6, se exponen las conclusiones y la exposición de trabajos actuales y futuros.

Causas y Costes del Mantenimiento del Software.

Los costes del MS tienen tendencia a crecer con el paso del tiempo. Algunos autores estiman que la situación puede llegar a ser casi insostenible. Existen empresas que se acercan a porcentajes del 95% de los recursos dedicados al mantenimiento, con lo cual se hace imposible el desarrollo de nuevos productos software. Esta situación se conoce como Barrera de Mantenimiento. Podemos afirmar que, en general, el porcentaje de recursos necesarios para mantenimiento se incrementa a medida que se produce más software [Hanna, 1993].

Son varias las causas de que, en la mayoría de las organizaciones actuales, se requiera mucho trabajo de mantenimiento. En primer lugar, una gran cantidad del software que existe actualmente ha sido desarrollado hace más de 10 años. Aunque estos programas fuesen creados utilizando las mejores técnicas de diseño y codificación existentes en su momento (la mayoría no lo fueron), se construyeron con restricciones de tamaño y espacio de almacenamiento y se desarrollaron con herramientas tecnológicamente desfasadas. En segundo lugar, estos programas han sufrido una o varias migraciones a nuevas plataformas o sistemas operativos. Y por último, han sufrido múltiples modificaciones para mejorarlos y adaptarlos a las nuevas necesidades de los usuarios. Todos estos cambios se realizaron sin tener en cuenta la arquitectura general del sistema (no se aplicaron técnicas de ingeniería inversa o reingeniería). El resultado de todo ello es la existencia de sistemas software, que tienen que seguir funcionando en la actualidad, con una baja calidad (diseño pobre de las estructuras de datos, mala codificación, lógica defectuosa y documentación escasa).

Una causa directa de los grandes costes del mantenimiento es que el coste relativo aproximado de reparar un defecto aumenta considerablemente en las últimas etapas del ciclo de vida del software [Piattini et al, 1998], 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).

Algunas de las razones por las que es menos costoso detectar y corregir un error durante las etapas iniciales del ciclo de vida que durante las etapas últimas son [Schach, 1992]:

- Es más fácil cambiar la documentación (por ejemplo, los documentos de especificación o de diseño) que modificar el código.
- Un cambio durante una fase tardía puede requerir que sea modificada la documentación de todas las fases anteriores.
- Es más fácil encontrar un defecto durante la fase en la cual se ha introducido el defecto que tratar de detectar y corregir los efectos provocados por el defecto en una fase posterior.

- La causa de un defecto puede esconderse en la inexistencia o falta de actualización de los documentos de especificación o diseño.

Además de los costes monetarios, el MS implica otros costes, menos tangibles que los primeros, pero que pueden ser causa de muchas preocupaciones. Un coste intangible del mantenimiento del software se encuentra en las oportunidades de desarrollo que se han de posponer o que se pierden, debido a que los recursos disponibles están dedicados a las tareas de mantenimiento. Otros costes intangibles son los siguientes:

- Insatisfacción del cliente cuando no se puede atender en un tiempo aceptable una petición de reparación o modificación que parece razonable.
- Los errores ocultos introducidos al cambiar el software durante el mantenimiento reducen la calidad global del producto.
- Perjuicio en otros proyectos de desarrollo cuando la plantilla tiene que dejarlos,

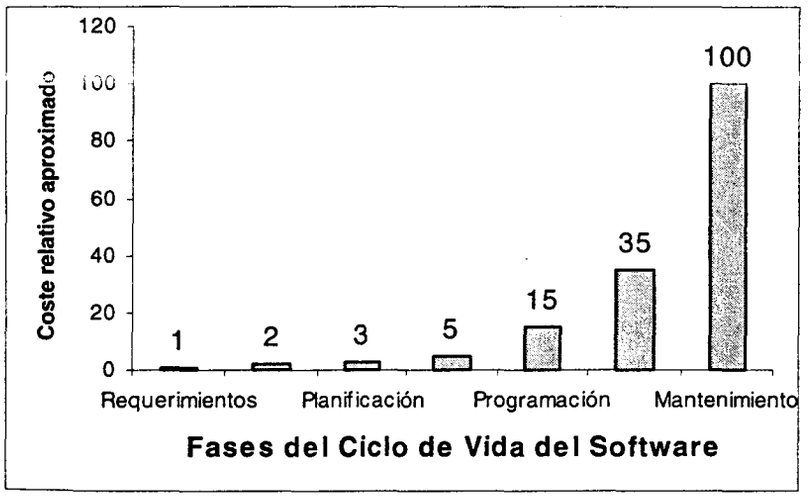


Figura 1. Coste relativo aproximado de detectar y corregir defectos

total o parcialmente, para atender peticiones de mantenimiento.

En suma, un coste final del mantenimiento del software es la reducción que se produce en la productividad de los informáticos cuando se inicia el mantenimiento de aplicaciones antiguas. Algunos autores [Pigoski, 1996] han calculado reducciones de la productividad - medida en 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.

El Proceso de Mantenimiento del Software (PMS).

En la norma ISO 12207, 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).

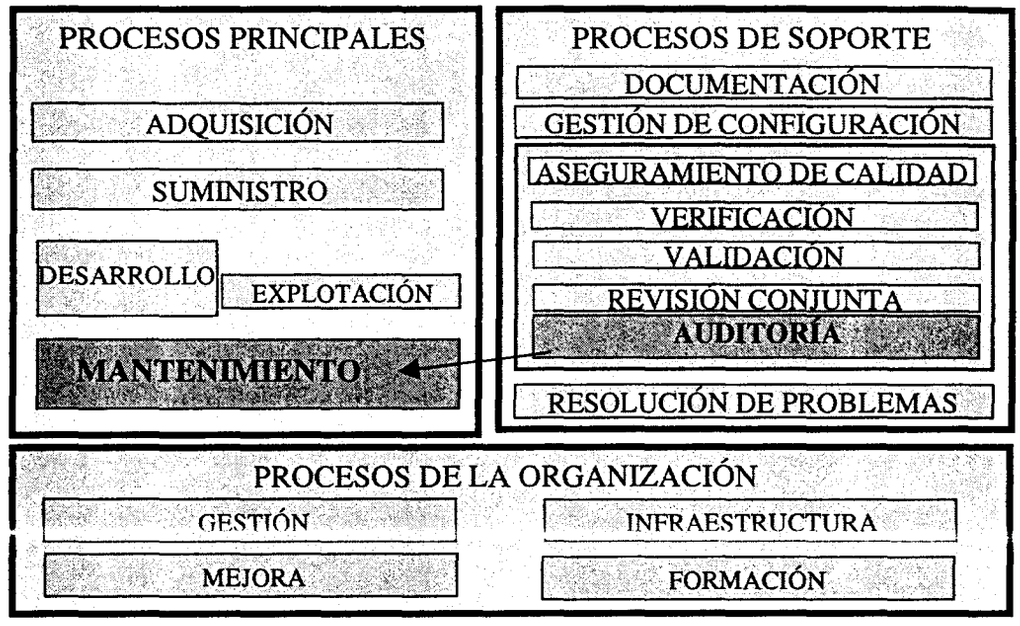


Figura 2 Estructura de procesos

El PMS incluye las 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 [Ruiz et al, 1999] y se establecen diversos aspectos del PMS que deberemos tener en cuenta al planificar la auditoría de dicho proceso.

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 (la persona, grupo u organización responsable del mantenimiento).

- Durante la *Implementación del Proceso,* el mantenedor:

  Desarrolla el plan y los procedimientos de mantenimiento (el *Plan de Mantenimiento* es un documento con la estrategia a seguir para realizar el mantenimiento);

  - Establece procedimientos para recibir, registrar y seguir la pista a los informes de problemas (PR) y requerimientos de modificación (MR) de los usuarios; y
  - Implementa o define los interfaces organizacionales con el proceso de gestión de configuraciones.

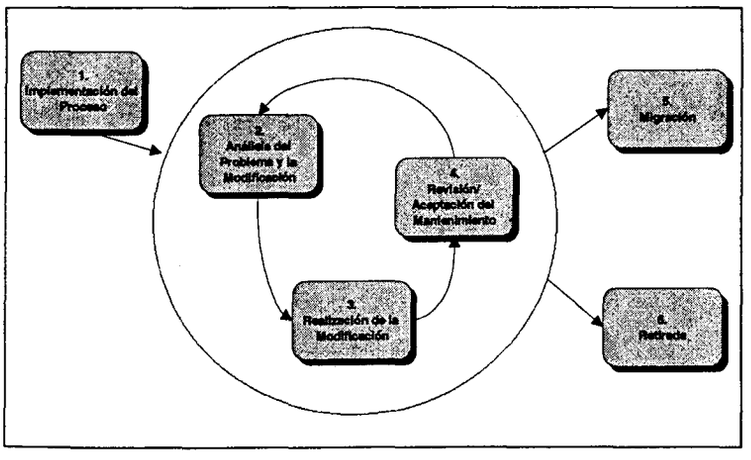


Figura 3. Actividades del Proceso de Mantenimiento del Software

- Durante la etapa de *Análisis del Problema y la Modificación*, el mantenedor:
  - Analiza el informe del problema o requerimiento de modificación para determinar su impacto en la organización, en el sistema existente y en los interfaces;
  - Replica o verifica el problema;
  - Define varias opciones para implementar la modificación;
  - Documenta el informe del problema o requerimiento de modificación, los resultados y opciones de implementación; y
  - Obtiene la aprobación para la opción de modificación seleccionada.

- Durante la etapa de *Realización de la Modificación*, el mantenedor:
  - Realiza un análisis para determinar los "elementos software" [17] que deben ser modificados; e
  - Invoca al proceso de desarrollo del software para realizar la modificación (incluyendo las pruebas).

- Durante la etapa de *Revisión/Aceptación del Mantenimiento*, el mantenedor:
  - Tiene entrevistas con la autoridad (cliente externo o interno) correspondiente para determinar la correcta integridad del sistema modificado; y
  - Obtiene la aprobación de la modificación mediante los mecanismos determinados previamente (en un contrato o similar).

- La etapa de *Migración* no es obligatoria, sólo existe cuando un producto software es modificado para funcionar en un nuevo entorno operativo. En ese caso, el mantenedor:
  - Diseña un plan de migración;
  - Notifica a los usuarios el inicio y conclusión de la migración;
  - Entrena a los usuarios en el nuevo entorno;
  - Evalúa el impacto del nuevo entorno; y
  - Archiva el producto software antiguo.

- La etapa de *Retirada* tampoco es obligatoria ya que sólo existe cuando un producto software ha concluido su vida útil y es sustituido por otro nuevo. En ese caso, el mantenedor realiza tareas similares a las referidas para la migración.

---

[17] *Se entiende como elemento software cualquier componente de un producto software: código fuente, ejecutable, documento de análisis, diagrama de diseño, manual, documentación de pruebas, etc.*

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

No es extraño leer en los medios de comunicación informaciones sobre empresas en las que habiendo sido objeto de auditoría y obtenido un informe favorable, poco después se detectan 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 [ISAFC, 1998] 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), 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 [Peña, 1998].

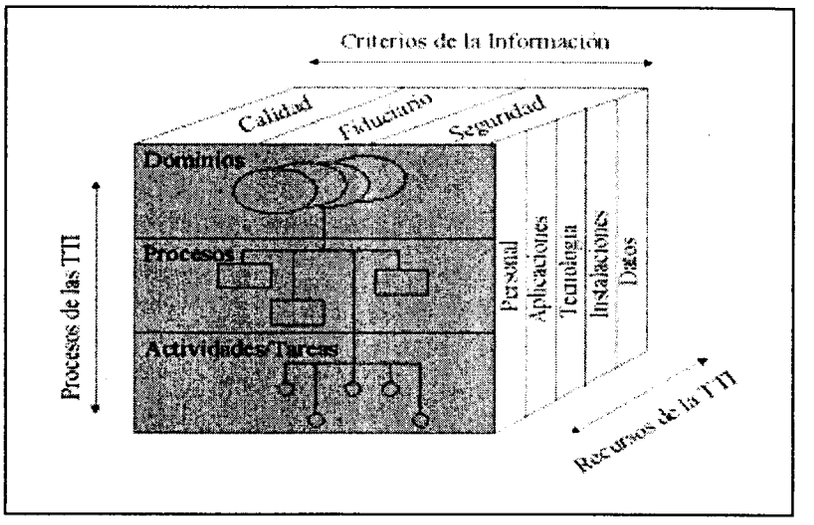


Figura 4. Las tres dimensiones conceptuales de CobiT

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 [COSO, 1994]: efectividad y eficiencia de las operaciones, fiabilidad de la información, legalidad (cumplimiento de las leyes y normas);
- Requisitos de seguridad: confidencialidad, integridad, disponibilidad.

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.

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.

Adaptación de CobiT al Proceso de Mantenimiento del Software.

Las fuentes bibliográficas más recientes dedicadas a la auditoría de sistemas de información [Weber, 1999], [Champlain, 1998], [Piattini y Peso, 1998] dedican muy poca o ninguna atención a la auditoría del MS. Ante esta situación, hemos ampliado la *Metodología MANTEMA para la Gestión Integral del Mantenimiento del Software* [Polo et al, 1999] con una propuesta para la auditoría del PMS, que presentamos a continuación.

En CobiT, los 34 OCGs propuestos se concretan en 302 objetivos de control detallados (OCDs). En la metodología MANTEMA 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 versión 2.0 de la metodología MANTEMA, 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 [18] 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.

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.

---

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

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

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 modifican y 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.

Referencias.

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

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

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

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

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

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

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

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

[Piattini et al, 1998] 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.

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

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

[Polo et al, 1999] 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.

[Ruiz et al, 1999] 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.

[Schach, 1992] Schach, S.R., *Practical Sofware Engineering*. Ed. Irwin & Aksen. USA 1992.

[Weber, 1999] Weber, R., *Information Systems Control and Audit*. Prentice-Hall. USA 1999.