

# Una propuesta de herramienta para administración y gerenciamiento del mantenimiento de proyectos de Software.

A.C. María Vanesa Amadeo<sup>1</sup>  
A.C. María Belén Alonso<sup>2</sup>  
Lic. Rodolfo Bertone<sup>3</sup>

L.I.D.I.<sup>4</sup>

*Departamento de Informática  
Facultad de Ciencias Exactas  
Universidad Nacional de La Plata*

## Resumen

Una de las etapas críticas en el ciclo de vida del software es el mantenimiento, en particular la administración de cambios. Controlar las modificaciones, involucra una serie de pasos que van desde manipular documentación realizada a partir del pedido de cambio, hasta la instalación de la nueva versión de software generada a partir de las modificaciones solicitadas. Esto incluye actividades que deben ser realizadas con los recursos humanos, de hardware y software existentes y además, actividades propias de la reingeniería.

El control de calidad no es ajeno a esta etapa. La forma en que se efectúa el gerenciamiento de los cambios es tenida en cuenta para determinar la calidad final de un producto.

El objetivo de este paper es presentar una propuesta de herramienta, que, utilizando la tecnología workflow, mejore la administración y desarrollo de cambios producidos, teniendo en cuenta para los mismos la documentación del proyecto y los recursos humanos, de hard y soft existentes en la empresa.

---

<sup>1</sup> Ayudante Diplomada, Becaria Ad-Honorem. LIDI E-Mail: [vanesaa@info.unlp.edu.ar](mailto:vanesaa@info.unlp.edu.ar)

<sup>2</sup> Alumna de Licenciatura en Informática, UNLP.

<sup>3</sup> Profesor Adjunto Dedicación Exclusiva, Cátedra Introducción a las Bases de Datos. E-Mail: [pbertone@info.unlp.edu.ar](mailto:pbertone@info.unlp.edu.ar)

<sup>4</sup> Laboratorio de Investigación y Desarrollo en Informática, Depto de Informática, Facultad de Ciencias Exactas, Universidad Nacional de La Plata. Calles 50 y 115 1er. Piso. (1900) La Plata. Buenos Aires. Argentina. TE-FAX: 54 21 227707.

## Introducción

El mantenimiento del software se caracteriza por dejar visible sólo una parte del problema existente y ocultar la magnitud real del mismo. Esto subyace con una gran cantidad de problemas potenciales y de costos. En la actualidad el mantenimiento del software existente puede llevarse hasta el 70% del esfuerzo para el desarrollo. [PRE93]

Los problemas de mantenimiento pueden ser provocados por diversos factores:

- *deficiencias en el análisis inicial de requisitos.* Una incorrecta comunicación con los usuarios lleva a un relevamiento incompleto de los objetivos que el software debe alcanzar. El producto final tendrá que entrar rápidamente en una etapa de mantenimiento a fin de intentar llegar a los objetivos que no fueron previstos. Este problema es fácilmente reducible si se logran avances en la etapa de Ingeniería de Requerimiento. [BAL97]
- *deficiencias en la construcción del software.* Son ejemplos de estos: un diseño incorrecto, una codificación inadecuada o una etapa de prueba no llevada a cabo correctamente. De la misma forma que en el caso anterior, si se sigue una metodología de trabajo adecuada los problemas resultantes tienden a disminuir. [BIA97]
- *deficiencias en la etapa de mantenimiento.* La etapa de mantenimiento en si puede producir más inconvenientes. Es decir, modificar el software sin una metodología adecuada puede introducir nuevos errores.
- *cambios en las necesidades del usuario.* Los proyectos de software no se caracterizan por ser estáticos. Los requerimientos iniciales varían a medida que el software va evolucionando. Por lo tanto, aún construyendo software siguiendo estrictamente una metodología y con un análisis de requerimientos exhaustivo y correcto, el producto resultante deberá ser modificado para adaptarlo a nuevas necesidades.

A continuación se detallan algunos aspectos particulares de la etapa de mantenimiento tenidos en cuenta, así como los lineamientos generales de la metodología workflow empleada para el planteo de una solución.

## Mantenimiento

El mantenimiento, la última fase del proceso de ingeniería del software, se lleva, actualmente, la mayor parte del presupuesto del proyecto. A medida que se desarrollan más sistemas, surge una tendencia molesta: la cantidad de esfuerzo y de recursos dedicados al mantenimiento crece. Ocurre cada vez más a menudo que las organizaciones de software lleguen a ver al mantenimiento como una barrera, la cual impide a la empresa embarcarse en nuevos proyectos.

Existen cuatro tipos de mantenimiento [RAM96]:

- *correctivo:* actúa para corregir errores que no han sido descubiertos antes de poner en uso el software,
- *adaptativo:* se aplica cuando los cambios del entorno externo precipitan las modificaciones del software,
- *perfectivo:* incorpora mejoras solicitadas por el usuario,
- *preventivo:* mejora la futura facilidad de mantenimiento y la fiabilidad como base para las futuras mejoras.

El mantenimiento correctivo está relacionado con las deficiencias planteadas anteriormente, primero con la construcción del software y segundo con el relevamiento inicial. El primer caso tiene que ver con errores de diseño o implementación que no fueron detectados en la etapa de prueba del sistema, y deben ser solucionados rápidamente. El segundo, mucho más complicado, el producto de software no cumple las necesidades del usuario, los cambios a efectuar son mucho mayores.

El mantenimiento adaptativo o perfectivo está relacionado con cambios en las necesidades del usuario. Las modificaciones a realizar no son tan traumáticas como las del caso anterior, ya que no reportan errores no encontrados y que deban tener una rápida solución. Los cambios son realizados en común acuerdo con el usuario y pueden llevarse a cabo en forma metodológica, lo cual tiende a asegurar un cambio gradual que no influya sobre las operaciones existentes del software. [BER96]

Existe una diferencia muy marcada entre el mantenimiento correctivo y el adaptativo o perfectivo. El mantenimiento correctivo debe resolverse rápidamente, puesto que, cuando el mismo se produce, el sistema no puede continuar operando. Los restantes, básicamente, deben comenzar por un estudio exhaustivo de su impacto sobre las especificaciones o requerimientos originales del sistema. Este estudio permite evaluar la viabilidad del cambio a producir dándole elementos al administrador del proyecto o al usuario final para decidir si las modificaciones solicitadas serán llevadas a la práctica.

En caso de llevar adelante los cambios es vital la documentación de los mismos (para cualquiera de los tipos de mantenimiento definidos). Todas las etapas de la ingeniería de software (análisis inicial de requerimientos, estudio de viabilidad, análisis de sistema, diseño, implementación y prueba) generaron documentación, la cual fue aprobada por el usuario. Esta etapa no es la excepción, cualquier cambio debe ilustrarse, documentarse y ser verificado con el usuario, haciéndole al modelo del sistema los cambios pertinentes. Esto usualmente se hace completando un documento conocido como solicitud de pedido de cambio del sistema.

La actividad de mantenimiento puede involucrar alguno, o todos los siguientes detalles [Your93]:

- añadir terminadores nuevos al diagrama de contexto o eliminar anteriores,
- añadir nuevos elementos a la lista de acontecimientos o eliminar otros,
- si el cambio es sustancial, puede modificarse la declaración de propósitos en el modelo ambiental,
- cambios en:
  - ⇒ los modelos de flujos de datos,
  - ⇒ modelos de entidad-relación
  - ⇒ modelos de transición de estados
- modificación y/o refinamiento de las especificaciones de procesos y el diccionario de datos,
- cambios en la interface hombre-máquina.

Los realizadores de los cambios tiene la obligación de escribir una declaración de impacto. Esto es, una declaración precisa y detallada de las modificaciones necesarias en la especificación del sistema para poder implementar el cambio propuesto.

## Workflow

El avance de la tecnología, la capacidad y facilidad de manejo de las PC actuales, lleva a que las organizaciones cuenten con una gran cantidad de información de todo tipo: documentos, hojas de cálculo, análisis, estadísticas, gráficos y diagramas. La utilización de correo electrónico y las redes de área local permiten compartir toda esta información, facilitando así el trabajo en grupo o cooperativo. [FER95]

Debido a que el volumen de actividades que se resuelven utilizando la computadora en una empresa aumenta considerablemente surge la necesidad, para los responsables de la misma, de monitorear cómo y cuándo se realiza cada trabajo. Existe un mayor interés en conocer los costos asociados a cada tarea (paso por paso) y los costos involucrados en el trabajo en si.

Un software de workflow adecuado puede ser capaz de ayudar en la tarea de administrar el trabajo de una organización. Un workflow es un sistema cuyos elementos son actividades vinculadas con otras a través de una relación de triggering y que, además, pueden ser disparadas por eventos externos. Estas actividades representan un proceso de negocios que comienza con una obligación y termina con su finalización [JOO95].

El workflow trata la coordinación del trabajo productivo en los procesos de negocios y sus características son:

- concentrarse en la interacción entre los actores (Individuos, grupos y/o máquinas) más que en las actividades, lo que indica que es una combinación de computer science y business science.
- se involucran diferentes tipos de personas y unidades de la organización.
- los límites pueden estar dentro y fuera de la organización.
- la integración de procedimientos y herramientas.

El principal beneficio de un sistema workflow radica en una mayor productividad personal y grupal, permitiendo la eliminación de los cuellos de botella y una mejor distribución del trabajo realizado en grupo.

Una herramienta workflow permite el manejo de documentos, facilitando el control de versionado y manteniendo los niveles de seguridad necesarios; o la automatización de tareas y procesos, controlando su interacción. [MAY94]

Cuando se modela con esta herramienta, las funciones básicas son: (1) conocer qué datos es necesario manipular para cada tarea, tales como localización, propietario, nivel de seguridad; (2) saber cuándo son necesarios, (3) determinar quién los utilizará y, por último, (4) controlar su finalización.

Para cumplir con lo anteriormente definido, un workflow debe responder a un sistema de trabajo estructurado, que permita construir una aplicación, monitorear su flujo de actividades y controlar tanto las tareas involucradas como los actores que las realizan.

## Objetivos

A partir de las especificaciones producidas por un grupo de trabajo anterior [RAM96], el objetivo general fue la definición final e implementación de un ambiente

que permita administrar y gerenciar los cambios de los proyectos de software. Para ello las tareas involucradas fueron:

- recepción inicial del pedido de cambio y evaluación inicial
- análisis de viabilidad a cargo de los diversos integrantes del proyecto
- planificación del cambio, en caso que este se produzca
- determinar el equipo realizador del cambio
- administrar la documentación existente y la generada
- administrar comunicaciones entre integrantes del proyecto
- administrar el versionado de los documentos, módulos y productos en función a los cambios producidos.

## **Especificación de una Solución**

### **Consideraciones iniciales**

Para la especificación de la solución se tuvieron en cuenta un conjunto de premisas, necesarias para la utilización de la herramienta:

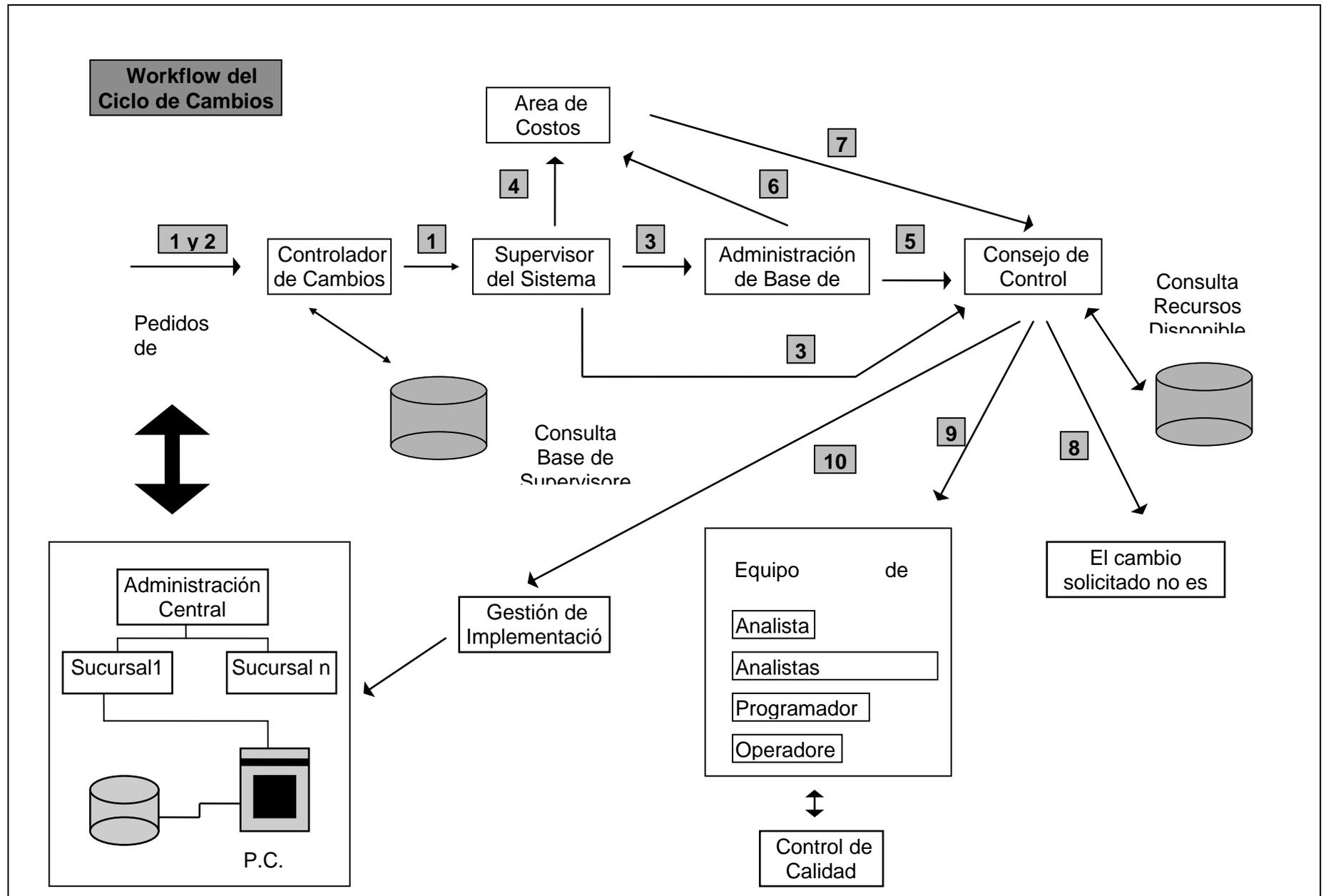
- La organización involucrada produce su propio software.
- Todos los desarrollos, son realizados utilizando una metodología de ingeniería de software.

El primer punto radica en que es vital disponer de toda la información generada durante cada etapa de la ingeniería de software. Si el desarrollo es interno (dentro de la propia empresa) esta condición debe darse sin mayores inconvenientes. Si el desarrollo fue externo la información generada por dichos desarrolladores también debería estar disponible.

El segundo punto requiere el uso de alguna metodología. Si el proyecto no fue desarrollado siguiendo pautas preestablecidas, es probable que no haya documentación o que, en caso de haberla, sea muy confusa o incompleta. Eso lleva a un caos en la etapa de mantenimiento que, con esta herramienta, no se puede solucionar.

### **Workflow de cambio**

La figura 1 muestra la propuesta para un workflow de cambios. En dicha figura se representan los distintos participantes del ciclo. Además se observa el flujo de tareas y la documentación necesaria para cada etapa. En la sección siguiente se describen cada una de las tareas asignadas a los participantes.



## Implementación de una solución

Para lograr un prototipo de implementación de la herramienta se necesitaron desarrollar cuatro tareas principales:

- ADMINISTRACIÓN DE LA HERRAMIENTA DE MANTENIMIENTO
- GESTIÓN DE LOS CAMBIOS
- IMPLEMENTACIÓN DEL CAMBIO
- DEFINICIÓN DE DOCUMENTACIÓN

La primer etapa consistió en definir los recursos (soft, hard y humanos) de la empresa y los proyectos administrados. La segunda y tercera administran el cambio desde que se recibe la solicitud hasta que se obtiene una solución. Por último, la cuarta etapa consistió en definir el formato de documentos necesario para administrar la información involucrada desde la generación del pedido de cambio hasta su resolución final.

### Administración de la herramienta de mantenimiento

Con este módulo se realizan las definiciones básicas, necesarias para el correcto funcionamiento de la herramienta. El Administrador se encarga de ingresar al CASE información general de la empresa y descripción particular de cada proyecto de software administrado. Estas tareas pueden resumirse en:

- definir datos generales de la empresa, como empleados, estadíos, roles, usuarios externos, etc. En particular, con los estadíos se define el ciclo que debe cumplir una solicitud de cambio desde que es generada hasta rechazarla o llevarla a cabo. Los roles definen para cada estadío la jerarquía del personal involucrado en ellos. Por ejemplo, el estadío de Administración de Base de Datos tiene un rol de director, otros de asistente, etc.
- descripción de cada proyecto. Esto incluye: datos particulares del proyecto, integrantes para cada estadío y que rol cumplen, ubicación y formato de documentos existentes o a crear; y por último, los usuarios del mismo.

Luego, en la etapa de gestión del cambio, se describirán cada uno de los estadíos posibles.

### Gestión de los cambios

Basados en la especificación de la solución propuesta, la cual está diagramada en la figura 1, el workflow de gestión de cambios está compuesto por una serie de estadíos. Cada uno de ellos recibe información de los anteriores y, basándose en esos datos y en su propio conocimiento del problema, genera la documentación que utilizará el estadío siguiente. Los siete estadíos definidos y sus objetivos son los siguientes:

- *Área de Control de Cambios*. Receptor del pedido de cambio. El controlador debe verificar la completitud del documento de pedido de cambios y la

autorización del mismo. En caso de no cumplir los requisitos definidos el documento es rechazado. La información generada en esta sección es enviada a la Supervisión del Sistema.

- *Supervisión del Sistema.* Recibe el pedido del cambio y el aval del controlador. En este estadio se dispone de información suficiente como para realizar una evaluación de la solicitud, permitiendo estimar los recursos necesarios (humanos, hard y soft) para el cambio. Se generan tres informes uno para el Área de Costos, otro para el Administrador de BD y el otro para el Consejo de Control.
- *Administración de Base de Datos.* A partir del informe recibido de la Supervisión del Sistema, realizan una evaluación de las modificaciones que serán necesarias efectuar sobre el modelo de datos existente, como afecta al sistema y los recursos necesarios de esta área para llevar adelante el cambio. Esta sección genera dos informes uno para el área de costos y otro para el Consejo de Control.
- *Área de Costos.* A esta sección llegan dos informes, de la supervisión y de la administración de BD. Basándose en ellos realiza un informe pormenorizado del impacto económico del cambio. Este informe es enviado al Consejo de Control para su posterior evaluación.
- *Consejo de Control.* Recibe informes sobre los distintos aspectos del pedido de cambio, generados por las áreas anteriores. Este grupo analiza toda la documentación recibida y determina si el cambio es viable o no. En caso de ser viable, elabora un plan de proyecto indicando tareas, plazos y equipo de proyecto responsable de llevar adelante el cambio. En caso contrario genera un informe donde se explica porqué el pedido de cambio es rechazado.
- *Equipo de Proyectos.* Es el encargado de implementar el pedido de cambio, siguiendo la metodología de Ingeniería de software utilizada en la empresa y respetando los parámetros de calidad establecidos. Entre estos parámetros se incluye la administración de las nuevas versiones de software y la documentación asociada a ellos, que incluye la información generada por el pedido de cambio.
- *Control de Instalación y versiones:* el último paso de la metodología está relacionado con administrar la instalación del nuevo producto. Esto debe hacerse con cronogramas establecidos y aprobados por los generadores del pedido de cambios y el equipo de control. Además, se debe realizar un control "fino" de la instalación, esto se relaciona con el hecho que la nueva versión de soft obtenida puede necesitar nuevos recursos, los cuales sólo estarán disponibles para algunos usuarios del producto final. Por lo tanto, en algunas dependencias puede estar utilizándose la nueva versión, en tanto que en otras, versiones anteriores.

## Implementación del Cambio

La herramienta prevé un seguimiento detallado de las actividades que comienzan con la aprobación del cambio. El Consejo de Control debe construir un Pert (como un diagrama de Gantt) determinando tiempos de ejecución, asignación de recursos (hard, soft y humanos) y responsables de la ejecución del pedido de cambio.

Esta sección involucra un sub-workflow dentro del workflow de administración del cambio. Este sub-workflow de resolución es el encargado de gerenciar la implementación del cambio. Entre las actividades que debe desarrollar están la

definición de los actores que lo compondrán cada una de las tareas: analistas, diseñadores, programadores, tanto senior como junior, administración de documentación y determinar además el equipo de prueba del producto.

Cada actor debe tener definida su jerarquía dentro del cambio. Los niveles de jerarquía propuestos son: director del cambio, encargado de tarea, integrante de grupo de tarea.

### **Definición de la documentación**

En la especificación de una solución se planteó la necesidad de trabajar con alguna metodología de Ingeniería de Software. Cualquiera de ellas plantea la necesidad de generación de documentación, la cual debe ser aprobada por el usuario. Esta documentación es necesariamente utilizada por los encargados de realizar mantenimiento.

Además, para cada solicitud de cambio se debe generar una serie de documentos que definan el pedido y el análisis que se realiza sobre el mismo. Como se observa en la figura 1, se plantea un conjunto de, a priori, 10 documentos. Se presenta a continuación una breve descripción de los mismos, el número indicado a la derecha se relaciona con los números de la figura 1:

- pedido de cambio (1): es el documento inicial, con él se genera una solicitud de modificación. Debe contener la naturaleza del cambio, una evaluación del tipo de cambio: nivel (error, mejora, o adaptación) y una prioridad. El mismo debe estar avalado por el responsable de la sección que lo solicita.
- autorización del cambio (2): el documento anterior es recibido por el controlador de cambios, y a partir de la solicitud confecciona este documento si los datos del pedido de cambio están completos.
- informe técnico por pedido de cambio (3): el supervisor del sistema recibe un pedido de cambio autorizado, a partir de él genera un informe técnico donde evalúa la incidencia del cambio, o sea qué efectos provocará el mismo. Dichos efectos pueden incidir sobre cualquier etapa del desarrollo. Puede afectar un módulo en particular, un conjunto de módulos, la especificación de diseño, la especificación del sistema, etc. Además, establece el nivel del cambio y la prioridad con que el mismo debe ser atendido, utilizando para ello las definiciones del usuario y su propia experiencia.
- informe técnico para el área de costos (4): el supervisor debe generar un informe indicando los recursos que serán necesarios para efectuar el cambio, el cual es enviado al área de costos para su posterior evaluación.
- informe del Administrador de Base de Datos por pedido de Cambio (5) : el administrador de Bases de datos realiza un análisis de los datos que se verán afectados por el cambio y confecciona este informe indicando qué otros sistemas los utilizan . Luego el consejo de control lo utilizará para evaluar los efectos laterales del cambio.
- informe del Administrador de Base de datos para el Área Costos (6) : el administrador de Base de datos además, analiza los recursos de personal especializado en Base de Datos, necesarios para la realización del cambio.
- informe del Área Costos por pedido de cambio (7): el área de costos realiza un informe, utilizando los documentos 4 y 6, sobre los recursos humanos y de base de datos necesarios para efectuar el cambio.

- informe de Pedido de Cambio rechazado (8): cuando un pedido de cambio es rechazado por el consejo de control , se genera este informe justificando adecuadamente las razones del rechazo.
- plan de trabajo (9): después que un pedido de cambio es aceptado, el consejo de control arma el equipo de proyecto y define este informe detallando los pasos a seguir y el tiempo asociado a cada uno para la realización del cambio.
- cronograma de Instalación (10): una vez terminada la modificación , la nueva versión debe ser instalada en las oficinas del usuario. En este informe se guarda un registro del plan de instalación esta versión.

## Conclusiones

La etapa de mantenimiento en el ciclo de vida del software fue una de las principales causantes de la crisis del software. Con esta herramienta se administran los pasos, documentos, y soft resultante involucrados en cada pedido de cambio reduciendo y controlando los efectos del mantenimiento.

Los beneficios que presenta la herramienta para una administración automatizada del mantenimiento son más sencillos de evaluar ante un cambio perfectivo o adaptativo. La administración de un cambio correctivo, podría suponerse demasiado burocrática ante la necesidad de una respuesta inmediata. Sin embargo la herramienta prevé este tipo de situaciones. Ante un cambio correctivo avalado por el controlador de cambios, la herramienta automáticamente acorta todos los plazos para el análisis del mismo. El Consejo de Control debe generar un plan de actividades acorde con la celeridad de resolución del cambio.

Además, el uso de nuevas metodologías de análisis y diseño, como por ejemplo JAD, genera la documentación correspondiente a la Ingeniería de Requerimientos, y posteriormente la generada en el diseño del sistema. Toda esta información es utilizada y actualizada durante la etapa de mantenimiento a fin de evaluar el pedido de modificación y decidir que acción tomar al respecto. El objetivo final de la herramienta es integrarla con JAD-CASE, prototipo que administran las etapas tempranas del ciclo de vida.

La utilización de esta herramienta ayudará a proteger la inversión realizada en el desarrollo del soft, aumentando su vida útil, ya que cambios bien administrados pueden aumentar la calidad y/o funcionalidad del mismo, sin hacerlo inestable. Además, resultará en un ahorro de los recursos humanos dedicados al mantenimiento, permitiendo un crecimiento en la productividad de las organizaciones.

## Bibliografía

- [BAL97] Una propuesta de workflow para la especificación de documentos para el proceso de ingeniería de requerimientos. Balda, Vicenzi, Bertone. ICIE 97. Facultad de Ingeniería. Buenos Aires 1997.
- [BER96] Herramienta para la administración de problemas y cambios. Bertone, Ramon, Russo, Gorga, Madoz. Encuentro de Ingeniería de Software. Santiago del Estero 1996.
- [BIA97] Apuntes de Master de automatización de Oficinas. Bianchi Alejandro. Dpto. Informática. Fac. Ciencias Exactas UNLP 1997.
- [FER95] Workflow. Jorge Ben Fernández, LAN & WAN, Abril 95.
- [JOO95] Fundamental Concepts for Workflow Automation in Practice. Stef Joosten and Sjaak Brinkkemper, Centre for Telematics and Information Technology, University of Twente, The Netherlands.
- [MAY94] Conozca las Herramientas de Flujo de Trabajo. Thorton A. May, BYTE, Julio 94, Mc. Graw Hill.
- [PRE93] Ingeniería de Software . Un enfoque práctico. Pressman Roger. McGrawHill. 1993
- [RAM96] Workflow para la Etapa del Mantenimiento de Software en el Ciclo de Vida Clásico. Ramon, Bertone, Russo, Gorga, Madoz. Anales del IV encuentro chileno de computación. Noviembre 1996.
- [YOU93] Análisis Estructurado Moderno. Edward Yourdon, Ed. Prentice Hall, 1993.