GUÍA PARA LA INSTALACIÓN E IMPLANTACIÓN DE UN NUEVO SOFTWARE

Resumen / ***Abstract***
Se propone una guía en la que se toman en cuenta los aspectos más generales a revisar al instalar e implantar un nuevo sistema. Cuando estos no se revisan de manera organizada, los resultados pueden ser muy negativos y las pérdidas considerables. La propuesta presentada fue puesta en práctica con éxito en una situación real en Finalse SA (Casa Financiera de la Corporación Cubalse).

A guide for installing and implanting a new softwareThe main objective of this article is proposing a guide with the most general aspects that should be considered before installing and implanting a new software. When these aspects are not checked in an organized manner, very negative consequences can be achieved resulting in considerable losses. The guide proposed was successfully used in a real environment in Finalse SA. (Financial House of the Cubalse Corporation).

Palabras clave ***/ Key words***
Instalación, implantación, ambiente de prueba

Installing, implanting, test environment

INTRODUCCIÓN

La **auditoría informática** crece en importancia a la par que la informática y los sistemas informáticos. Esto ocurre en la misma medida en que aumenta la necesidad de las entidades de disponer de adecuados sistemas de información.[1]

Una de las tantas definiciones de **auditoría informátic**a es la siguiente:[2]

Conjunto de procedimientos y técnicas que permiten en una entidad: evaluar, total o parcialmente, el grado en que se cumplen la observancia de los controles internos asociados al sistema informático; determinar el grado de protección de sus activos y recursos; verificar si sus actividades se desarrollan eficientemente y de acuerdo con la normativa informática y general existentes en la entidad, y para conseguir la eficacia exigida en el marco de la organización correspondiente.

El principio de practicar auditorías a los sistemas informáticos con el uso de herramientas informáticas, se debe a varios factores. Entre ellos se encuentran el desarrollo vertiginoso de la informática y la existencia de un elevado grado de aplicaciones de procesamiento de datos orientados a la gestión; unido a la necesidad de dotar a las organizaciones de un instrumento de control que promueva una beneficiosa expectativa a un costo razonable y eleve constantemente el control interno.

Practicar auditorías en un ambiente computarizado, donde la informatización de los sistemas de gestión y contables han alcanzado un desarrollo tan notable, supone la introducción de una concepción muy diferente a la que primó durante décadas; con nuevos enfoques en los que inexcusablemente, la informática tiene que participar activamente como una valiosísima herramien-

Mercedes Almuiña Fernández, *Licenciada en Ciencias de la Computación, Especialista en Computación, Casa Financiera de la Corporación Cubalse, (Finalse SA), Ciudad de La Habana, Cuba*
e-mail:mercy@cubals.cu

Roberto Sepúlveda Lima, *Ingeniero Electricista, Doctor en Ciencias Técnicas, Profesor Titular, Centro de Estudios de Ingeniería de Sistemas (CEIS), Instituto Superior Politécnico José Antonio Echeverría, Cujae, Ciudad de La Habana, Cuba*
e-mail:sepul@ceis.cujae.edu.cu

Recibido:Enero del 2005
Aprobado:Marzo del 2005

ta, que permita a esta disciplina evolucionar al mismo ritmo de las transformaciones incorporadas a la estructura del registro y el control interno.[1]

DESARROLLO

Auditoría interna

El Instituto de Auditores Internos (The Institute of Internal Auditors, Inc-IIA) la definen así:

> La auditoría interna es una función independiente de evaluación establecida dentro de una organización, para examinar y evaluar sus actividades como un servicio a la organización. El objetivo de la auditoría interna consiste en apoyar a los miembros de la organización en el desempeño de sus responsabilidades. Para ello la auditoría interna les proporciona, análisis, evaluaciones, recomendaciones, asesoría e información concerniente con las actividades revisadas.[3]

Las ventajas de la auditoría interna radican en que esta puede actuar periódicamente realizando revisiones globales, como parte de su plan anual y de su actividad normal. Las recomendaciones benefician el trabajo de la empresa.

Herramientas actuales para realizar auditorías. Técnicas de auditoría asistidas por computadora (TAAC)

Cuando el profesional del control utiliza tradicionalmente la computadora,[2] como una herramienta amistosa, muchos de sus problemas actualmente complejos tales como el volumen de los datos, la complejidad de su estructura, la relación e interdependencia entre diferentes fuentes, pueden ser manejadas y enfrentadas con dominio más preciso y profesional.

Hoy, gracias a los softwares especializados de auditoría, es posible efectuar auditorías al 100 % de las evidencias. (Entiéndase bases de datos, archivos de incidencias, registros, bitácoras, datos.) De allí que la aplicación de técnicas para el muestreo estadístico, que también se ha simplificado gracias a estas herramientas, se deja más para definiciones a nivel inicial de la auditoría, aspectos macros, pero no para el desarrollo de pruebas sustantivas, es decir, pruebas de datos ya que el revisar el 100 % de las evidencias no encierra dificultad ni complejidad a través del proceso automatizado, entregan mayor confianza y certeza en las evidencias obtenidas.

Las TAACs incrementan o amplían el alcance de la investigación y permiten realizar pruebas que no pueden efectuarse manualmente. Al incrementar el alcance y calidad de los muestreos, verifican un gran número de elementos. Por todo lo anterior, elevan la calidad y fiabilidad de las verificaciones a realizar y a su vez garantizan el menor número de interrupciones posibles a la entidad auditada.

El software WINIDEA,[4] es uno de los dos primeros softwares generales de auditoría, por su posición en el mercado internacional.[1] No obstante, es muy reconocido el software ACL (Audit Control Language: Software integrado que provee control del acceso de datos, administración, análisis y presentación. Consultar sitio http://www.acl.com/), y en la actualidad es la herramienta que cuenta con la mayor cuota del mercado de las auditorías y es el instrumento oficial en otros países, para el análisis de los datos o para la también denominada auditoría de las transacciones.

Al respecto, en Cuba se ha difundido y se ha capacitado a profesionales en el software IDEA,[4] en sus versiones para Windows.[5] En otros países, se prepara a los profesionales en softwares como ACL, Wizrule (Aplicación para la auditoría de datos y limpieza. Analiza las bases de datos y muestra las inconsistencias en los mismos. Consultar sitio http://www.wizsoft.com/rule.html), Galileo (Sistema de módulos completamente integrado que puede ser ajustado a las necesidades específicas de un auditor interno o un departamento orientado a proyectos. Consultar sitio http://www.galileoontheweb.com/).

Descripción de algunas herramientas

IDEA (Interactive Data Extraction and Analysis)

IDEA es una poderosa herramienta,[4] para auditores, contadores y administradores financieros que necesitan revisar, analizar, manipular, extraer y evaluar información contenida en sistemas, bases de datos y archivos electrónicos.

IDEA permite la ejecución de procesos como consultas a archivos de datos, calcular totales o promedios, encontrar si las transacciones o registros cumplen un criterio dado o buscar campos inusuales.

COBIT (Governance, Control and Audit for Information and Related Technology)

En la elaboración de esta guía fue necesario apoyarse principalmente en bibliografía sobre auditoría y el estándar COBIT. Este se está usando en Cuba en entidades como ETECSA y CIMEX.

La **misión** de COBIT es la siguiente: "Investigar, desarrollar, publicar y promover un conjunto de objetivos de control en tecnología de información con autoridad, actualizados, de carácter internacional y aceptados generalmente para el uso cotidiano de gerentes de empresas y auditores".[6]

COBIT está diseñado,[7] como un estándar aplicable y aceptable en general para la buena práctica de la auditoría de las tecnologías de la información en todo el mundo. El producto COBIT utiliza los objetivos de control de ISACA, mejorados con estándares específicos de tipo técnico, profesional, normativo e industrial existentes y emergentes. Los objetivos de control se han desarrollado para su aplicación en el amplio espectro de sistemas de información en la empresa de prestaciones importantes (normas, reglas, etcétera).

Aspectos a auditar

Luego de efectuar la investigación de las herramientas actuales y de estudiar la bibliografía actual relacionada con el tema, se obtuvo una serie de metodologías y controles útiles para revisar algunos aspectos, pero no se encontró una que abarcara en su

totalidad todos los requerimientos que se querían. Algunas de las vulnerabilidades de los sistemas no pueden ser encontradas por ningún auditor de código, por lo que se requiere que sean auditados localmente por personas familiarizadas con el mismo.[8] De ahí que se impuso la necesidad de elaborar una guía.

A continuación se muestra un pequeño resumen con los aspectos más relevantes a tomar en cuenta al auditar internamente un software.

Los elementos a tomar en cuenta fueron:

- Entrada de datos.
- Procesamiento.
- Salida de datos.
- Integridad de datos.
- Autenticación de usuarios.
- Acceso al sistema.
- Protección contra ataques externos.
- Respaldo del sistema.
- Segregación de funciones.
- No repudio.
- Prueba en implantación.

El primer aspecto a tomar en cuenta se refiere a la **entrada de datos**. Las interfaces internas y externas se deben diseñar y documentar apropiadamente. Los procedimientos de manejo de errores durante la creación de datos deberán asegurar razonablemente que los errores y las irregularidades sean detectados, reportados y corregidos.[9]

Muchos son los aspectos a tener en cuenta para el **procesamiento**; se deben establecer procedimientos de manejo de errores en el procesamiento de datos que permitan la identificación de transacciones erróneas sin que estas sean procesadas y sin interrumpir el procesamiento de otras transacciones válidas. La aplicación debe contar con la existencia de procesos automáticos para las operaciones fundamentales. Se deben crear **alertas** (del inglés Alerts) del sistema[10] y definir siempre un **operador** (del inglés Operators)**,** para ser notificado cuando ocurren errores de determinado nivel de severidad. A su vez, se deben hacer pruebas de operadores para asegurarse de que estos reciban las notificaciones. Por otra parte, debe existir el **banco de datos** y estar correctamente conformado.

Los requerimientos de **salida de datos**,[2,9] deben estar bien documentados. En el caso de reportes con información sensible se deben destruir inmediatamente las salidas de las corridas abortadas. Después de un análisis global de reportes, es importante determinar si hay algunos que puedan ser eliminados, combinados, agrupados, simplificados, o si se requieren, nuevos reportes. Se deben incluir los siguientes elementos en cada reporte: fecha de preparación, periodo cubierto de proceso, título descriptivo del contenido del reporte, departamento usuario, número de identificación del programa.

El tema de la **integridad de los datos** es uno de los más importantes.[11] Se deben usar los **tipos de datos** (del inglés Data Types) más exactos posibles.[12] Todas las tablas que conforman las bases de datos deben tener **llave primaria** (del inglés Primary Key) correctamente definida.[13] Igualmente, deben tener **llaves foráneas** (del inglés Foreign Key) definidas para ser usadas como restricciones de integridad al establecerse las relaciones entre las tablas. Es fundamental verificar la existencia de **diagrama de entidad relación**,[13] y que las **bases de datos** estén correctamente diseñadas. Las actualizaciones y eliminaciones en las tablas se deben realizar de forma que se mantenga la integridad referencial .[12]

Se deben definir e implementar apropiados procedimientos para asegurar:[12] **Atomicidad** (unidad de trabajo indivisible, todas sus acciones tienen éxito o todas fallan),**Consistencia** (si la transacción no logra alcanzar un estado final estable, deberá regresar al sistema a su estado inicial), **Durabilidad** (los efectos de una transacción son permanentes después que concluye su proceso), **Aislamiento** (el comportamiento de una transacción no es afectado por otras transacciones que se ejecutan concurrentemente).

Son generalmente más conocidos los aspectos relacionados con la Seguridad. Estos se asocian en su mayoría con la **Autenticación de usuarios** y el **acceso al sistema**.[9]

Son los propios usuarios quienes deben controlar en forma sistemática la actividad de sus propias cuentas.[14] Estos deben tener permisos de acceso a **vistas** (del inglés Views) y **procedimientos almacenados** (del inglés Stored Procedures) y no a las tablas. Los **inicios de sesión** (del inglés login) se deben limitar a ciertas horas del día.[15] Es conocida la importancia de que no se usen contraseñas en blanco y que cumplan la característica de ser complejas.

Por otra parte, el **acceso al sistema** debe encontrarse limitado mediante el empleo de palabras claves y estas a su vez controlar el acceso de personas responsabilizadas a las operaciones fundamentales. Es importante que la información contenida en el fichero de palabras claves se encuentre cifrada.[16]

La copia de seguridad de los programas de la aplicación, debe guardarse fuera del local,[9] donde está ubicada la computadora. El plan de recuperación debe demostrar que la organización de forma anticipada ha valorado todos los posibles riesgos de seguridad y ha desarrollado procedimientos para la recuperación de estos.

Otro aspecto importante es la **protección del sistema contra ataques externos**.[17]

El **respaldo del sistema** es otro tema muy amplio.[7] Garantizar el estado consistente después de respaldar una base de datos y después de restaurarla, es fundamental; así como chequear los respaldos de base de datos periódicamente.[10] Es importante tener información sobre los respaldos. Medida muy importante y generalmente olvidada es respaldar también las tablas del sistema, al igual que programar las operaciones de respaldo cuando la actividad de la base de datos es baja.

Para proteger la información deben estar generados los **archivos de comandos** (del inglés Scripts) de los fuentes relevantes.[18]

Otros aspectos generales a tomar en cuenta son los **cambios en el sistema**,[9] la segregación de funciones,[9] administración de problemas, [9] y el no repudio.[9]

Los cambios deben ser aprobados mediante solicitudes escritas y documentados internamente. Debe elaborarse un **plan de gestión de configuración**.[19] Los programas para prueba final se deben almacenar en directorios separados. Finalmente, no se debe olvidar actualizar la documentación.

Para lograr la **segregación de funciones**[7] los analistas y programadores no deben tener acceso a la operación. De la misma forma, los operadores no deben tener acceso libre a las bibliotecas ni a los lugares donde se tengan los datos almacenados.

Relacionado con el **no repudio**, es importante garantizar que se esté almacenando suficiente información cronológica en bitácoras de operaciones para permitir la reconstrucción y la revisión de las secuencias de tiempo de procesamiento. La aplicación debe contar con procedimientos programados (trazas) que permitan conocer la ocasión en que las personas autorizadas hacen uso de la misma y las funciones que emplean.[9]

La última etapa es la de **prueba e implantación del sistema**.[7]

Deben cumplirse los requerimientos mínimos del sistema. Analizando el rendimiento del mismo, se pueden predecir los recursos requeridos.[10] El personal de los departamentos usuarios y el grupo de operaciones deberán estar entrenados de acuerdo con el plan de entrenamiento definido.[9] Las pruebas piloto o en paralelo deben ser llevadas a cabo de acuerdo con un plan establecido y los criterios para la terminación del proceso de pruebas, especificados con anterioridad.

El sistema debe validarse como un producto completo antes de ponerlo en operación.[7]

APLICACIÓN DE LA GUÍA PROPUESTA

La guía antes expuesta fue aplicada en Finalse SA, para auditar el sistema financiero SABIC.

El **Sistema Informatizado para la Banca Internacional de Comercio** (SABIC) fue desarrollado por el Banco Central de Cuba, con el fin de satisfacer las necesidades de procesamiento de datos de los bancos pequeños y medianos que operen en este ámbito.

La versión existente anteriormente era la versión en MS-DOS, la cual fue desarrollada en FOXPRO para plataforma MS -DOS, en una red NOVELL versión 3.12. Se implantó una nueva versión, basada en la técnica cliente-servidor, y que utiliza Visual FoxPro como cliente y SQL Server 2000 como servidor sobre una plataforma Windows 2000.[20]

CONCLUSIONES

Durante la realización del trabajo, los auditores se encuentran con nuevas tecnologías de avanzada, por lo que requieren de la incorporación sistemática de herramientas actuales y de conocimientos cada vez más profundos en la materia. A pesar de la gran cantidad de herramientas existentes es difícil encontrar una que cumpla con todos los requerimientos deseados,[21,22] de ahí la importancia de crear metodologías más personalizadas.

La guía propuesta satisface los requisitos de seguridad e integridad, que constituyen los requerimientos que se necesitaban en el contexto de los sistemas estudiados.

La aplicación de la guía propuesta ha sido de gran utilidad en Finalse SA, Como consecuencia de ello, se analizaron las debilidades del sistema y se ejecutaron acciones en aras de cumplir los objetivos propuestos al modernizar el sistema informático existente anteriormente.

RECOMENDACIONES

Se recomienda a los desarrolladores de software y a las instituciones que tomen en cuenta los aspectos abordados en este artículo. La guía propuesta debe aplicarse antes de implantar un nuevo software o uno significativamente modificado. Para consultar la guía en su totalidad se aconseja dirigirse a la Tesis de Maestría de la autora que se discutirá próximamente.

Cualquier modificación o crítica puede contribuir a perfeccionamiento las acciones efectuadas por los autores.

REFERENCIAS

1. **RIVAS, CATALINA:** *La auditoría en el contexto actual*. Con sultado en el sitio http://www.gestiopolis.com/recursos/ docu mentos/fulldocs/fin/auditcontxactual.htm en octubre 2004. Gestiopolis.com fue creado en el año 2000.
2. **ZAVARO BABANI, LEÓN Y CEFERINO MARTÍNEZ GARCÍA:** *Auditoría Informática*, Ciudad de La Habana, 1999.
3. Boletin Ausideas: "Ideas para considerar en el control interno y la auditoría de su Empresa", septiembre, 2003.
4. Pricewa Terhou Secooper: *Qué es IDEA*, consultado en el sitio http://www.pwc.co.cr/ITS/IDEA%20-BROCHURE.pdf en octubre 2004.
5. **MÉRIDA MUÑOZ, JORGE Y GUILLERMO WOOD FONSECA:** "Curso de Auditoria con Informática", Ponencia presentada en el VI Seminario Iberoamericano de Seguridad en Tecnologías de Información y Comunicaciones, celebrado en el Palacio de las Convenciones, Ciudad de La Habana, 2002.
6. Information Systems Audit and Control Foundation**:** "Cobit, Governance, Control And Audit For Information And Related Technology", Second Edition, 1998.
7. Information Systems Audit and Control Foundation: "Cobit, Governance, Control And Audit For Information And Related Technology", Third Edition, 1999.
8. **HEFFLEY, JOHN AND PASCAL MEUNIER:** "Can Source Code Auditing Software Identify Common Vulnerabilities

and Be Used to Evaluate Software Security?", Proceedings of the 37th Hawaii International Conference on System Sciences, 2004. Tomado del sitio de IEEE Computer Society.

9. **VANDAMA ESTÉVEZ, NANCY Y MILAGROS LESCAY CORDERO**: "Implementación del modelo Cobit en el desarrollo de las Auditorías de Sistemas de Información", 2002, Ponencia presentada en el VI Seminario Iberoamericano de Seguridad en Tecnologías de Información y Comunicaciones, celebrado en el Palacio de las Convenciones.
10. Microsoft Training and Certification: Administering a Microsoft SQL Server 2000 Database", 2000.
11. **DAVIDSON, LOUIS:** "Professional SQL Server 2000 Database Design", Wrox Press, 200.1
12. **GUERRERO, FERNANDO G. Y CARLOS EDUARDO ROJAS:** *Programación en Microsoft SQL Server 2000 con ejemplos*, 1ra. ed., Pearson Education SA, diciembre 2001,
13. **BYRNE, JEFFRY L.:** "Microsoft SQL Server, what database administrators need to know", 1997
14. Microsoft Training and Certification: "Designing a Secure Microsoft Windows 2000 Network", 2000.
15. http:// www.respondanet.com/spanish/anti_corrupcion/informes/Guatemala98/AUD_Aredes.doc en octubre 2004. Sitio del Proyecto de Responsabilidad / Anti-corrupción en las Américas, creado en 1998.
16. **SEPÚLVEDA LIMA, ROBERTO; FRANK DAVID ABÁ Y LESTER CRESPO:** "Escenarios típicos de fallas de seguridad relacionadas con la integridad de datos", *Ingeniería Industrial,* Vol XXIV, No 3. pp. 87, Ciudad de La Habana, Cuba, 2003.
17. **ANCAJIMA ROJAS, ROBERT G.:** "Seguridad en SQL Server" , septiembre 2004, consultado en el sitio http:// www.informatizate.net del grupo Informatízate, on-line desde el 27 de noviembre del 2002.
18. Microsoft Training and Certificatio: "Programming a Microsoft SQL Server 2000 Database", 2000.
19. **VILLARROEL ACEVEDO, RODOLFO:** *Mejora miento del proceso de gestión de configuración de software*, abril, 2004. Consultado en http://alarcos.inf-cr.uclm.es/doc/software/rodolfo.pdf, sitio del Grupo Alarcos de la Escuela Superior de Informática de Ciudad Real.
20. *Manual de Usuario Sabic*, Banco Central de Cuba: 2004.
21. **JIWNANI, K. AND M. ZELKOWITZ:** "Maintaining Software with a Security Perspective", International Conference on Software Maintenance, octubre 2002, Canada. Tomado del sitio de IEEE Computer Society.
22. **CHANGCHIT, CHULEEPORN; CLYDE HOLSAPPLE AND DONALD MADDEN:** "Positive Impacts of an Intelligent System on Internal Control Problem Recognition", Proceedings of the 32th Hawaii International Conference on System Sciences, 1999. Tomado del sitio de IEEE Computer Society.