**EL ROL DEL ANALISTA DE SISTEMAS.**

Desde hace mucho tiempo, las organizaciones han reconocido la importancia de administrar recursos clave como la mano de obra y las materias primas. En la actualidad, la información se ha ganado el legítimo derecho de ser considerada como un recurso clave. Los encargados de la toma de decisiones por fin han comprendido que la información no es tan sólo un producto derivado de la conducción de los negocios, sino un impulsor de los mismos y que puede constituir un factor crucial en el éxito o fracaso de una empresa.

Para maximizar la utilidad de la información, una empresa debe administrarla de manera eficiente, como lo hace con los demás recursos. Los administradores deben comprender que los costos tienen una estrecha relación con la producción, distribución, seguridad, almacenamiento y recuperación de toda la información. A pesar de que la información está en todas partes, no es gratuita, y no se debe asumir que se podrá usar estratégicamente para aumentar la competitividad de una empresa.

**TIPOS DE SISTEMAS**

Los sistemas de información se desarrollan con diversos propósitos, según las necesidades de la empresa. Los sistemas de procesamiento de transacciones funcionan al nivel operativo de una organización, los sistemas de automatización de la oficina y los sistemas de trabajo del conocimiento apoyan el trabajo al nivel del conocimiento. Los sistemas de información gerencial y los sistemas de apoyo a la toma de decisiones se encuentran entre los sistemas de alto nivel. Los sistemas expertos aplican el conocimiento de los encargados de la toma de decisiones para solucionar problemas estructurados específicos. Los sistemas de apoyo a ejecutivos se encuentran en el nivel estratégico de la administración. Los sistemas de apoyo a la toma de decisiones en grupo y los sistemas de trabajo corporativo apoyados por computadora, descritos de manera más general, auxilian la toma de decisiones semiestructuradas o no estructuradas a nivel de grupo.

Dentro de los tipos de sistemas están:

***SISTEMAS DE PROCESAMIENTO DE TRANSACCIONES***

Los sistemas de procesamiento de transacciones (TPS, *Transaction Processing Systems*) son sistemas de información computarizada creados para procesar grandes cantidades de datos relacionadas con transacciones rutinarias de negocios, como las nóminas y los inventarios. Un TPS elimina el fastidio que representa la realización de transacciones operativas necesarias y reduce el tiempo que una vez fue requerido para llevarlas a cabo de manera manual, aunque los usuarios aún tienen que capturar datos en los sistemas computarizados.

***SISTEMAS DE AUTOMATIZACIÓN DE LA OFICINA Y SISTEMAS DE TRABAJO DEL CONOCIMIENTO***

Existen dos clases de sistemas en el nivel del conocimiento de una organización. Los sistemas de automatización de la oficina (OAS, *Office Automation Systems*) apoyan a los trabajadores de datos, quienes por lo general no generan conocimientos nuevos, sino más bien analizan la información con el propósito de transformar los datos o manipularlos de alguna manera antes de compartirlos o, en su caso, distribuirlos formalmente con el resto de la organización y en ocasiones más allá de ésta. Entre los componentes más comunes de un OAS están el procesamiento de texto, las hojas de cálculo, la autoedición, la calendarización electrónica y las comunicaciones mediante correo de voz, correo electrónico y videoconferencia.

Los sistemas de trabajo del conocimiento (KWS, *Knowledge Work Systems*) sirven de apoyo a los trabajadores profesionales, como los científicos, ingenieros y médicos, en sus esfuerzos de creación de nuevo conocimiento y dan a éstos la posibilidad de compartirlo con sus organizaciones o con la sociedad.

***SISTEMAS DE INFORMACIÓN GERENCIAL***

Los sistemas de información gerencial (MIS, *Management Information Systems*) no reemplazan a los sistemas de procesamiento de transacciones, más bien, incluyen el procesamiento de transacciones. Los MIS son sistemas de información computarizados cuyo propósito es contribuir a la correcta interacción entre los usuarios y las computadoras. Debido a que requieren que los usuarios, el software (los programas de cómputo) y el hardware (las computadoras, impresoras, etc.), funcionen de manera coordinada, los sistemas de información gerencial dan apoyo a un espectro de tareas organizacionales mucho más amplio que los sistemas de procesamiento de transacciones, como el análisis y la toma de decisiones.

***SISTEMAS DE APOYO A LA TOMA DE DECISIONES***

Los sistemas de apoyo a la toma de decisiones (DSS, *Decision Support Systems*) constituyen una clase de alto nivel de sistemas de información computarizada. Los DSS coinciden con los sistemas de información gerencial en que ambos dependen de una base de datos para abastecerse de datos. Sin embargo, difieren en que el DSS pone énfasis en el apoyo a la toma de decisiones en todas sus fases, aunque la decisión definitiva es responsabilidad exclusiva del encargado de tomarla. Los sistemas de apoyo a la toma de decisiones se ajustan más al gusto de la persona o grupo que los utiliza que a los sistemas de información gerencial tradicionales. En ocasiones se hace referencia a ellos como sistemas que se enfocan en la inteligencia de negocios.

***SISTEMAS EXPERTOS E INTELIGENCIA ARTIFICIAL***

La inteligencia artificial (AI, *Artificial Intelligence*) se puede considerar como el campo general para los sistemas expertos. La motivación principal de la AI ha sido desarrollar máquinas que tengan un comportamiento inteligente. Dos de las líneas de investigación de la AI son la comprensión del lenguaje natural y el análisis de la capacidad para razonar un problema hasta su conclusión lógica. Los sistemas expertos utilizan las técnicas de razonamiento de la AI para solucionar los problemas que les plantean los usuarios de negocios (y de otras áreas).

***SISTEMAS DE APOYO A LA TOMA DE DECISIONES EN GRUPO Y SISTEMAS***

***DE TRABAJO COLABORATIVO APOYADOS POR COMPUTADORA***

Cuando los grupos requieren trabajar en conjunto para tomar decisiones semiestructuradas o no estructuradas, un sistema de apoyo a la toma de decisiones en grupo (GDSS, *Group* *Decision Support System*) podría ser la solución. Este tipo de sistemas, que se utilizan en salones especiales equipados con diversas configuraciones, faculta a los miembros del grupo a interactuar con apoyo electrónico —casi siempre software especializado— y la asistencia de un facilitador especial. Los sistemas de apoyo a la toma de decisiones en grupo tienen el propósito de unir a un grupo en la búsqueda de la solución a un problema con la ayuda de diversas herramientas como los sondeos, los cuestionarios, la lluvia de ideas y la creación de escenarios.

***SISTEMAS DE APOYO A EJECUTIVOS***

Cuando los ejecutivos recurren a la computadora, por lo general lo hacen en busca de métodos que los auxilien en la toma de decisiones de nivel estratégico. Los sistemas de apoyo a ejecutivos (ESS, *Executive Support Systems*) ayudan a estos últimos a organizar sus actividades relacionadas con el entorno externo mediante herramientas gráficas y de comunicaciones, que por lo general se encuentran en salas de juntas o en oficinas corporativas personales. A pesar de que los ESS dependen de la información producida por los TPS y los MIS, ayudan a los usuarios a resolver problemas de toma de decisiones no estructuradas, que no tienen una aplicación específica, mediante la creación de un entorno que contribuye a pensar en problemas estratégicos de una manera bien informada.

**INTEGRACIÓN DE LAS TECNOLOGÍAS DE SISTEMAS**

A medida que se adopten y difundan las nuevas tecnologías, parte del trabajo de los analistas de sistemas se dedicará a la integración de los sistemas tradicionales con los nuevos. Aquí describiremos algunas de las nuevas tecnologías de información que los analistas de sistemas utilizarán para empresas que buscan integrar sus aplicaciones de comercio electrónico con sus negocios tradicionales, o bien, iniciar negocios electrónicos completamente nuevos.

***APLICACIONES DE COMERCIO ELECTRÓNICO Y SISTEMAS WEB***

Muchos de los sistemas que se describen en este libro pueden dotarse de una mayor funcionalidad si se migran a la World Wide Web o si desde su concepción se implementan como tecnologías basadas en la Web. En una encuesta reciente la mitad de todas las empresas pequeñas y medianas respondieron que Internet fue su estrategia preferida para buscar el crecimiento de sus negocios. Esta respuesta duplicó a la de aquellos que manifestaron su inclinación por realizar alianzas estratégicas como medio para crecer. Hay muchos beneficios derivados de la implementación de una aplicación en la Web:

1. Una creciente difusión de la disponibilidad de un servicio, producto, industria, persona o grupo.

2. La posibilidad de que los usuarios accedan las 24 horas.

3. La estandarización del diseño de la interfaz.

4. La creación de un sistema que se puede extender a nivel mundial y llegar a gente en lugares remotos sin preocuparse por la zona horaria en que se encuentren.

***SISTEMAS DE PLANEACIÓN DE RECURSOS EMPRESARIALES***

Muchas organizaciones consideran los beneficios potenciales que se derivan de la integración de los diversos sistemas de información que existen en los diferentes niveles administrativos, con funciones dispares. Esta integración es precisamente el propósito de los sistemas de planeación de recursos empresariales (ERP, *Enterprise Resource Planning*). El establecimiento de los sistemas ERP implica un enorme compromiso y cambio por parte de la organización.

Es común que los analistas de sistemas desempeñen el papel de asesores en los proyectos de ERP que utilizan software patentado. Entre el software más conocido de ERP se encuentran SAP, PeopleSoft y paquetes de Oracle y J.D. Edwards. Algunos de estos paquetes están diseñados para migrar a las empresas a la Web. Por lo general, los analistas y algunos usuarios requieren capacitación, apoyo técnico y mantenimiento por parte del fabricante para diseñar, instalar, dar mantenimiento, actualizar y utilizar de manera apropiada un paquete de ERP en particular.

***SISTEMAS PARA DISPOSITIVOS INALÁMBRICOS Y PORTÁTILES***

Los analistas tienen la exigencia de diseñar una gran cantidad de nuevos sistemas y aplicaciones,

muchos de ellos para dispositivos inalámbricos y computadoras portátiles como la popular serie de computadoras Palm y otros asistentes personales digitales (PDAs, *Personal Digital Assistants*). Además, los analistas podrían llegar a diseñar redes de comunicaciones estándar o inalámbricas que integren voz, vídeo y correo electrónico en intranets para unaorganización o extranets para la industria. El comercio electrónico inalámbrico se conocecomo comercio móvil o m-commerce.

Las redes inalámbricas de área local (WLANs,*Wireless Local Area Networks*), las redes de fidelidad inalámbrica, conocidas como WI-FI, y las redes inalámbricas personales que agrupan a muchos tipos de dispositivos dentro del estándar conocido como Bluetooth, constituyen sistemas cuyo diseño podrían solicitarle a usted en su función de analista.

***SOFTWARE DE CÓDIGO ABIERTO***

El software de código abierto es una alternativa al desarrollo de software tradicional cuyo código patentado se oculta a los usuarios. Representa un modelo de desarrollo y filosofía de distribución de software gratuito y publicación de su código fuente. Bajo este esquema, el código (las instrucciones para la computadora) se puede estudiar y compartir, y muchos usuarios y programadores tienen la posibilidad de modificarlo. Las convenciones que rigen a esta comunidad incluyen que todas las modificaciones que se hagan a un programa deben compartirse con todos aquellos que participan en el proyecto. Entre los ejemplos se encuentran el sistema operativo Linux y el software Apache empleado en servidores que alojan sitios Web.

**LA NECESIDAD DEL ANÁLISIS Y DISEÑO DE SISTEMAS**

El análisis y diseño de sistemas, tal como lo realizan los analistas de sistemas, tiene el propósito de analizar sistemáticamente la entrada o el flujo de datos, procesar o transformar datos, el almacenamiento de datos y la salida de información en el contexto de una empresa en particular. Más aún, el análisis de sistemas se emplea para analizar, diseñar e implementar mejoras en el funcionamiento de las empresas, a través de sistemas de información computarizados. La instalación de un sistema sin una planeación adecuada conduce a una gran decepción y con frecuencia provoca que el sistema deje de utilizarse. El análisis y diseño de sistemas da forma al análisis y diseño de sistemas de información, un esfuerzo muy valioso que de otra manera podría haberse realizado de una manera fortuita. Se le puede considerar como una serie de procesos sistemáticamente emprendidos con el propósito de mejorar un negocio con ayuda de sistemas de información computarizados. Gran parte del análisis y diseño de sistemas implica trabajar con usuarios actuales y ocasionales de los sistemas de información.

**ROLES DEL ANALISTA DE SISTEMAS**

El analista de sistemas evalúa de manera sistemática el funcionamiento de un negocio mediante el examen de la entrada y el procesamiento de datos y su consiguiente producción de información, con el propósito de mejorar los procesos de una organización. Muchas mejoras incluyen un mayor apoyo a las funciones de negocios a través del uso de sistemas de información computarizados. Esta definición pone énfasis en un enfoque sistemático y metódico para analizar —y en consecuencia mejorar— lo que sucede en el contexto específico creado por un negocio. El analista debe tener la capacidad de trabajar con todo tipo de gente y contar con suficiente experiencia en computadoras. El analista desempeña diversos roles, en ocasiones varios de ellos al mismo tiempo. Los tres roles principales del analista de sistemas son el de consultor, experto en soporte técnico y agente de cambio.

***EL ROL DE CONSULTOR DEL ANALISTA DE SISTEMAS***

Con frecuencia, el analista de sistemas desempeña el rol de consultor para un negocio y, por tanto, podría ser contratado de manera específica para enfrentar los problemas de sistemas de información de una empresa. Esta contratación se puede traducir en una ventaja porque los consultores externos tienen una perspectiva fresca de la cual carecen los demás miembros de una organización. También se puede traducir en una desventaja porque alguien externo nunca conocerá la verdadera cultura organizacional. En su función de consultor externo, usted dependerá en gran medida de los métodos sistemáticos que se explican en este libro para analizar y diseñar sistemas de información apropiados para una empresa en particular.

Además, tendrá que apoyarse en los usuarios de los sistemas de información para entender la cultura organizacional desde la perspectiva que tienen ellos.

***EL ROL DE EXPERTO EN SOPORTE TÉCNICO DEL ANALISTA DE SISTEMAS***

Otro rol que tendrá que desempeñar es el de experto en soporte técnico dentro de la empresa en la cual labora de manera regular. En este rol el analista recurre a su experiencia profesional con el hardware y software de cómputo y al uso que se le da en el negocio. Con frecuencia, este trabajo no implica un proyecto completo de sistemas, sino más bien la realización de pequeñas modificaciones o la toma de decisiones que se circunscriben a un solo departamento.

***EL ROL DE AGENTE DE CAMBIO DEL ANALISTA DE SISTEMAS***

El rol más completo y de mayor responsabilidad que asume el analista de sistemas es el de agente de cambio, ya sea interno o externo para la empresa. Como analista, usted es un agente de cambio si desempeña cualquiera de las actividades relacionadas con el ciclo de vida del desarrollo de sistemas (que se explicará en la siguiente sección) y está presente en la empresa durante un largo periodo (de dos semanas a más de un año). Un agente de cambio se puede definir como alguien que sirve de catalizador para el cambio, desarrolla un plan para el cambio y coopera con los demás para facilitar el cambio.

***CUALIDADES DEL ANALISTA DE SISTEMAS***

Por lo visto anteriormente, se deduce fácilmente que el analista exitoso debe contar con una amplia gama de cualidades.

Hay una gran diversidad de personas trabajando como analistas de sistemas, por lo que cualquier descripción que intente ser general está destinada a quedarse corta en algún sentido.

No obstante, la mayoría de los analistas de sistemas tienen algunas cualidades comunes.

En primer lugar, el analista es un solucionador de problemas. Es una persona que **aborda como un reto** el análisis de problemas y que disfruta al diseñar soluciones factibles.

Cuando es necesario, el analista debe contar con la **capacidad de afrontar** sistemáticamente cualquier situación mediante la correcta aplicación de herramientas, técnicas y su experiencia.

El analista también debe ser un **comunicador con capacidad para relacionarse con los** demás durante extensos periodos. Necesita **suficiente experiencia** en computación para programar, entender las capacidades de las computadoras, recabar los requisitos de información de los usuarios y comunicarlos a los programadores. Asimismo, debe tener una ética personal y profesional firme que le ayude a moldear las relaciones con sus clientes.

El analista de sistemas debe ser una **persona autodisciplinada y automotivada,** con la capacidad de administrar y coordinar los innumerables recursos de un proyecto, incluyendo a otras personas. La profesión de analista de sistemas es muy exigente; pero es una profesión en constante evolución que siempre trae nuevos retos.

**EL CICLO DE VIDA DEL DESARROLLO DE SISTEMAS**

A lo largo de este capítulo, nos hemos referido al enfoque sistemático que el analista toma en relación con el análisis y diseño de sistemas de información. Gran parte de este enfoque se incluye en el ciclo de vida del desarrollo de sistemas (SDLC, *Systems Development Life* *Cycle*). El SDLC es un enfoque por fases para el análisis y el diseño cuya premisa principal consiste en que los sistemas se desarrollan mejor utilizando un ciclo específico de actividades del analista y el usuario.

Los analistas no se ponen de acuerdo en la cantidad de fases que incluye el ciclo de vida del desarrollo de sistemas, pero en general alaban su enfoque organizado. A pesar de que cada fase se explica por separado, nunca se realiza como un paso aislado. Más bien, es posible que varias actividades ocurran de manera simultánea, y algunas de ellas podrían repetirse. Es más práctico considerar que el SDLC se realiza por fases (con actividades en pleno apogeo que se traslapan con otras hasta terminarse por completo) y no en pasos aislados.

***IDENTIFICACIÓN DE PROBLEMAS, OPORTUNIDADES Y OBJETIVOS***

En esta primera fase del ciclo de vida del desarrollo de sistemas, el analista se ocupa de identificar problemas, oportunidades y objetivos. Esta etapa es crítica para el éxito del resto del proyecto, pues a nadie le agrada desperdiciar tiempo trabajando en un problema que no era el que se debía resolver.

La primera fase requiere que el analista observe objetivamente lo que sucede en un negocio.

Con frecuencia los problemas son detectados por alguien más, y ésta es la razón de la llamada inicial al analista. Las oportunidades son situaciones que el analista considera susceptibles de mejorar utilizando sistemas de información computarizados. El aprovechamiento de las oportunidades podría permitir a la empresa obtener una ventaja competitiva o establecer un estándar para la industria.

La identificación de objetivos también es una parte importante de la primera fase. En primer lugar, el analista debe averiguar lo que la empresa trata de conseguir. A continuación, podrá determinar si algunas funciones de las aplicaciones de los sistemas de información pueden contribuir a que el negocio alcance sus objetivos aplicándolas a problemas u oportunidades específicos.

***DETERMINACIÓN DE LOS REQUERIMIENTOS DE INFORMACIÓN***

La siguiente fase que enfrenta el analista es la determinación de los requerimientos de información de los usuarios. Entre las herramientas que se utilizan para determinar los requerimientos de información de un negocio se encuentran métodos interactivos como las entrevistas, los muestreos, la investigación de datos impresos y la aplicación de cuestionarios; métodos que no interfieren con el usuario como la observación del comportamiento de los encargados de tomar las decisiones y sus entornos de oficina, al igual que métodos de amplio alcance como la elaboración de prototipos.

El desarrollo rápido de aplicaciones (RAD, *Rapid Application Development*) es un enfoque orientado a objetos para el desarrollo de sistemas que incluye un método de desarrollo (que abarca la generación de requerimientos de información) y herramientas de software.

***ANÁLISIS DE LAS NECESIDADES DEL SISTEMA*** La siguiente fase que debe enfrentar el analista tiene que ver con el análisis de las necesidades del sistema. De nueva cuenta, herramientas y técnicas especiales auxilian al analista en la determinación de los requerimientos. Una de estas herramientas es el uso de diagramas de flujo de datos para graficar las entradas, los procesos y las salidas de las funciones del negocio en una forma gráfica estructurada. A partir de los diagramas de flujo de datos se desarrolla un diccionario de datos que enlista todos los datos utilizados en el sistema, así como sus respectivas especificaciones. Durante esta fase el analista de sistemas analiza también las decisiones estructuradas que se hayan tomado. Las decisiones estructuradas son aquellas en las cuales se pueden determinar las condiciones, las alternativas de condición, las acciones y las reglas de acción. Existen tres métodos principales para el análisis de decisiones estructuradas: español estructurado, tablas y árboles de decisión. ***DISEÑO DEL SISTEMA RECOMENDADO*** En la fase de diseño del ciclo de vida del desarrollo de sistemas, el analista utiliza la información recopilada en las primeras fases para realizar el diseño lógico del sistema de información. El analista diseña procedimientos precisos para la captura de datos que aseguran que los datos que ingresen al sistema de información sean correctos. Además, el analista facilita la entrada eficiente de datos al sistema de información mediante técnicas adecuadas de diseño de formularios y pantallas. La concepción de la interfaz de usuario forma parte del diseño lógico del sistema de información. La interfaz conecta al usuario con el sistema y por tanto es sumamente importante. Entre los ejemplos de interfaces de usuario se encuentran el teclado (para teclear preguntas y respuestas), los menús en pantalla (para obtener los comandos de usuario) y diversas interfaces gráficas de usuario (GUIs, *Graphical User Interfaces*) que se manejan a través de un ratón o una pantalla sensible al tacto. ***DESARROLLO Y DOCUMENTACIÓN DEL SOFTWARE*** En la quinta fase del ciclo de vida del desarrollo de sistemas, el analista trabaja de manera conjunta con los programadores para desarrollar cualquier software original necesario. Entre las técnicas estructuradas para diseñar y documentar software se encuentran los diagramas de estructura, los diagramas de Nassi-Shneiderman y el pseudocódigo. El analista se vale de una o más de estas herramientas para comunicar al programador lo que se requiere programar. Durante esta fase el analista también trabaja con los usuarios para desarrollar documentación efectiva para el software, como manuales de procedimientos, ayuda en línea y sitios Web que incluyan respuestas a preguntas frecuentes (FAQ, *Frequently Asked Questions*) en archivos “Léame” que se integrarán en el nuevo software. La documentación indica a los usuarios cómo utilizar el software y lo que deben hacer en caso de que surjan problemas derivados de este uso. ***PRUEBA Y MANTENIMIENTO DEL SISTEMA*** Antes de poner el sistema en funcionamiento es necesario probarlo. Es mucho menos costoso encontrar los problemas antes que el sistema se entregue a los usuarios. Una parte de las pruebas las realizan los programadores solos, y otra la llevan a cabo de manera conjunta con los analistas de sistemas. Primero se realiza una serie de pruebas con datos de muestra para determinar con precisión cuáles son los problemas y posteriormente se realiza otra con datos reales del sistema actual. El mantenimiento del sistema de información y su documentación empiezan en esta fase y se llevan a cabo de manera rutinaria durante toda su vida útil. Gran parte del trabajo habitual del programador consiste en el mantenimiento, y las empresas invierten enormes sumas de dinero en esta actividad. Parte del mantenimiento, como las actualizaciones de programas, se pueden realizar de manera automática a través de un sitio Web. Muchos de los procedimientos sistemáticos que el analista emplea durante el ciclo de vida del desarrollo de sistemas pueden contribuir a garantizar que el mantenimiento se mantendrá al mínimo.

***IMPLEMENTACIÓN Y EVALUACIÓN DEL SISTEMA***

Ésta es la última fase del desarrollo de sistemas, y aquí el analista participa en la implementación del sistema de información. En esta fase se capacita a los usuarios en el manejo del sistema. Parte de la capacitación la imparten los fabricantes, pero la supervisión de ésta es responsabilidad del analista de sistemas. Además, el analista tiene que planear una conversión gradual del sistema anterior al actual. Este proceso incluye la conversión de archivos de formatos anteriores a los nuevos, o la construcción de una base de datos, la instalación de equipo y la puesta en producción del nuevo sistema.

Se menciona la evaluación como la fase final del ciclo de vida del desarrollo de sistemas principalmente en aras del debate. En realidad, la evaluación se lleva a cabo durante cada una de las fases. Un criterio clave que se debe cumplir es si los usuarios a quienes va dirigido el sistema lo están utilizando realmente.

***IMPACTO DEL MANTENIMIENTO***

Después de instalar un sistema, se le debe dar mantenimiento, es decir, los programas de cómputo tienen que ser modificados y actualizados cuando lo requieran. Según estimaciones, los departamentos invierten en mantenimiento de 48 a 60 por ciento del tiempo total del desarrollo de sistemas. Queda muy poco tiempo para el desarrollo de nuevos sistemas. Conforme se incrementa el número de programas escritos, también lo hace la cantidad de mantenimiento que requieren.

El mantenimiento se realiza por dos razones. La primera es la corrección de errores del software. No importa cuán exhaustivamente se pruebe el sistema, los errores se cuelan en los programas de cómputo. Los errores en el software comercial para PC se documentan como “anomalías conocidas”, y se corrigen en el lanzamiento de nuevas versiones del software o en revisiones intermedias. En el software hecho a la medida, los errores se deben corregir en el momento que se detectan.

La otra razón para el mantenimiento del sistema es la mejora de las capacidades del software en respuesta a las cambiantes necesidades de una organización, que por lo general tienen que ver con alguna de las siguientes tres situaciones:

1. Con frecuencia, después de familiarizarse con el sistema de cómputo y sus capacidades, los usuarios requieren características adicionales.

2. El negocio cambia con el tiempo.

3. El hardware y el software cambian a un ritmo acelerado.

**USO DE HERRAMIENTAS CASE**

A lo largo de este libro hacemos énfasis en la necesidad de un enfoque sistemático e integral para el análisis, diseño e implementación de sistemas de información. Reconocemos que para ser productivos, los analistas de sistemas deben realizar sus tareas de una manera organizada, precisa y minuciosa. Desde principios de la década de 1990, los analistas empezaron a beneficiarse de las herramientas de productividad, denominadas herramientas de Ingeniería de Software Asistida por Computadora (CASE, *Computer-Aided Software Engineering*), que se crearon explícitamente para mejorar su trabajo rutinario mediante apoyo automatizado.

De acuerdo con un estudio reciente, era más probable que los departamentos de sistemas de información con más de 10 empleados adoptaran las herramientas CASE que los departamentos con menos empleados. Los sistemas, procedimientos y prácticas administrativas de las organizaciones podrían restringir la difusión de las herramientas CASE. Los analistas de sistemas se apoyan en estas herramientas, desde el principio hasta el fin del ciclo de vida, para incrementar la productividad, comunicarse de manera más eficiente con los usuarios e integrar el trabajo que desempeñan en el sistema.

***RAZONES PARA EL USO DE LAS HERRAMIENTAS CASE***

**Aumento en la productividad del analista** Visible Analyst (VA) es una herramienta CASE que da al analista de sistemas la posibilidad de realizar planeación, análisis y diseño por medios gráficos, con el propósito de construir aplicaciones cliente-servidor y bases de datos complejas. Esta herramienta permite modelar los datos, procesos y objetos en diferentes formatos.

Visible Analyst genera información sobre el modelo en muchas formas distintas, incluyendo COBOL, C, Visual Basic, SQL y XML.

**Mejora de la comunicación analista-usuario** Para que el sistema propuesto se concrete y sea útil en la práctica, es esencial una excelente comunicación entre analistas y usuarios durante todo el ciclo de vida del desarrollo de sistemas. El éxito de la futura implementación del sistema depende de la capacidad de analistas y usuarios para comunicarse de una manera eficiente. Hasta el momento, de las experiencias de analistas que utilizan herramientas CASE se desprende que su uso fomenta una mayor y más eficiente comunicación entre usuarios y analistas.

**Integración de las actividades del ciclo de vida** La tercera razón para el uso de las herramientas

CASE es integrar las actividades y proporcionar continuidad de una fase a la siguiente durante todo el ciclo de vida del desarrollo de sistemas.

**Evaluar de manera precisa los cambios en el mantenimiento** La cuarta, y probablemente una de las razones más importantes para el uso de herramientas CASE, es que permiten a los usuarios analizar y evaluar el impacto de los cambios en el mantenimiento. Por ejemplo, el tamaño de un elemento como un número de cliente podría requerir alargarse. La herramienta CASE pueden generar referencias cruzadas de cada pantalla, informe y archivo en el cual sea utilizado el elemento, dando lugar a un plan de mantenimiento integral.

**HERRAMIENTAS CASE DE BAJO Y ALTO NIVEL**

Las herramientas CASE se clasifican como de bajo nivel, de alto nivel e integradas, estas últimas combinando las de alto y bajo nivel en un solo conjunto. A pesar de que los expertos difieren en los criterios que definen con precisión cuáles son las herramientas CASE de alto nivel y cuáles las de bajo nivel, podría ser útil clasificarlas con base en los usuarios a los que dan apoyo. Las herramientas CASE de alto nivel ayudan principalmente a los analistas y diseñadores, en tanto que las de bajo nivel son utilizadas con más frecuencia por programadores y trabajadores que deben implementar los sistemas diseñados con herramientas CASE de alto nivel.

***HERRAMIENTAS CASE DE ALTO NIVEL***

Una herramienta CASE de alto nivel da al analista la posibilidad de crear y modificar el diseño del sistema. Toda la información relacionada con el proyecto se almacena en una enciclopedia denominada depósito CASE, una enorme colección de registros, elementos, diagramas, pantallas, informes e información diversa. Con la información del depósito se podrían generar informes que muestren dónde está incompleto el diseño o dónde contiene errores.

Las herramientas CASE de alto nivel también pueden apoyar la modelación de los requerimientos funcionales de una organización, ayudar a los analistas y usuarios a definir el alcance de un proyecto determinado y a visualizar la forma en que el proyecto se combina con otras partes de la organización. Además, algunas herramientas CASE de alto nivel pueden ayudar en la creación de prototipos de diseños de pantallas e informes.

***HERRAMIENTAS CASE DE BAJO NIVEL***

Las herramientas CASE de bajo nivel se utilizan para generar código fuente de computadora, eliminando así la necesidad de programar el sistema. La generación de código tiene varias ventajas:

1. El sistema se puede generar más rápido que si se tuviera que escribir todos los programas.

No obstante, con frecuencia el periodo para familiarizarse con la metodología utilizada por el generador de código es muy largo, por lo que la generación del programa podría ser más lenta al principio. Además, es necesario ingresar por completo el diseño en el conjunto de herramientas, tarea que podría tomar un tiempo considerable.

2. La generación de código reduce el tiempo invertido en el mantenimiento. No hay necesidad de modificar, probar y depurar los programas de computadora. En lugar de eso, al modificar el diseño CASE se vuelve a generar el código. Si se invierte menos tiempo en el mantenimiento, se tiene más tiempo para desarrollar nuevos sistemas y aligerar la acumulación de proyectos en espera de desarrollo.

3. Más de un lenguaje de computadora, de tal manera que se facilita la migración de sistemas de una plataforma, digamos de *mainframe,* a otra, como una PC. Por ejemplo, la edición de VA para corporaciones puede generar código fuente en lenguajes de tercera generación como ANSI, COBOL o C.

4. La generación de código ofrece una forma económica de ajustar los sistemas comerciales de fabricantes de sistemas a las necesidades de la organización. Con frecuencia, la modificación de esta clase de software implica un esfuerzo tan grande que su costo es mayor al de la compra del mismo. Con el software de generación de código, la compra de un diseño CASE y un depósito CASE para la aplicación permite al analista modificar el diseño y generar el sistema de cómputo modificado.

5. El código generado está libre de errores de programación. Los únicos errores potenciales son los de diseño, los cuales se pueden minimizar produciendo informes de análisis

CASE para garantizar que el diseño del sistema esté completo y correcto.

**INGENIERÍA INVERSA Y REINGENIERÍA DE SOFTWARE**

La ingeniería inversa y la reingeniería de software son métodos para alargar la vida de programas anteriores, conocidos como software heredado. En ambos métodos se emplea software de reingeniería asistida por computadora (CARE, *Computer-Assisted Reengineering*) para analizar y reestructurar el código de computadora existente. En el mercado hay varios conjuntos de herramientas de ingeniería inversa.

Observe que el término *reingeniería* se utiliza en numerosos contextos diferentes de ingeniería, programación y negocios. Con frecuencia se emplea para denotar “reingeniería de procesos de negocios”, que es una forma de darle una nueva orientación a los procesos clave de una organización. Los analistas de sistemas pueden desempeñar un rol importante en la reingeniería de procesos de negocios, puesto que muchos de los cambios requeridos sólo se pueden lograr mediante el uso de tecnología de información novedosa.

La ingeniería inversa es lo opuesto a la generación de código. El código fuente de la computadora es examinado, analizado y convertido en entidades para el depósito. El primer paso de la ingeniería inversa de software es cargar, en el conjunto de herramientas, el código de programa existente (tal como se haya escrito en COBOL, C o cualquier otro lenguaje de alto nivel). Según el conjunto de herramientas de ingeniería inversa que se utilice, el código es analizado y las herramientas producen algunos o todos los elementos siguientes:

1. Estructuras de datos y elementos que describen los archivos y registros almacenados por el sistema.

2. Diseños de pantallas, si el programa es en línea.

3. Esquemas de informes para programas por lotes.

4. Un diagrama de estructura que muestra la jerarquía de los módulos del programa.

5. Diseño y relaciones de bases de datos.

El diseño almacenado en el depósito podría modificarse o incorporarse en información de otro proyecto CASE. Cuando se terminan todas las modificaciones, el nuevo código del sistema puede volver a generarse. La reingeniería se refiere al proceso completo de convertir el código de programa al diseño CASE, modificar el diseño y volver a generar el nuevo código de programa.

Son varias las ventajas que se consiguen al utilizar un conjunto de herramientas de ingeniería inversa:

1. Reducción del tiempo requerido para el mantenimiento del sistema, con lo cual queda más tiempo para nuevos desarrollos.

2. Se genera documentación, que podría haber sido inexistente o mínima en los programas anteriores.

3. Se crean programas estructurados a partir de código de computadora no estructurado o pobremente estructurado.

4. Los cambios futuros al mantenimiento son más sencillos, porque se pueden realizar al nivel del diseño más que al nivel del código.

5. Es posible analizar el sistema con el fin de eliminar porciones sin utilizar de código de computadora, el cual aún podría estar presente en programas anteriores a pesar de que las revisiones hechas al programa a lo largo de los años lo hayan vuelto obsoleto.

***ANÁLISIS Y DISEÑO DE SISTEMAS ORIENTADO A OBJETOS***

El análisis y diseño orientado a objetos es un enfoque cuyo propósito es facilitar el desarrollo de sistemas que deben cambiar con rapidez en respuesta a entornos de negocios dinámicos.

Es difícil trabajar bien con técnicas orientadas a objetos en situaciones en las cuales sistemas de información complicados requieren mantenimiento, adaptación y rediseño de manera continua. Los enfoques orientados a objetos utilizan el estándar de la industria para la modelación de sistemas orientados a objetos, el lenguaje unificado de modelación (UML, *Unified Modeling Language*), para analizar un sistema en forma de modelo de casos de uso.

La programación orientada a objetos difiere de la programación tradicional de procedimientos en que la primera examina los objetos que conforman un sistema. Cada objeto es una representación en computadora de alguna cosa o suceso real. Los objetos pueden ser clientes, artículos, pedidos, etc. Los objetos se representan y agrupan en clases, que son óptimas para su reutilización y mantenimiento. Una clase define el conjunto de atributos y comportamientos que comparten los objetos que ésta contiene.

**PROGRAMACIÓN EXTREMA Y OTRAS METODOLOGÍAS ALTERNAS**

Aunque lo anteriormente tratado se enfoca en la metodología que actualmente se utiliza de manera más amplia, en ocasiones el analista tendrá que reconocer que la organización se podría beneficiar de una metodología alterna. Quizá un proyecto de sistemas con un enfoque estructurado haya fallado, o quizá las subculturas que existen en la organización, compuestas por diferentes grupos de usuarios, parezcan más proclives a utilizar un método alterno. En un espacio tan breve no podríamos analizar adecuadamente estos métodos alternos, que merecen y han sido explicados en sus propios libros e investigaciones. Sin embargo, al mencionarlos aquí esperamos que usted tome conciencia de que, bajo ciertas circunstancias, su organización podría requerir una alternativa o complemento para un análisis y diseño estructurado y para el ciclo de vida del desarrollo de sistemas.

La programación extrema (XP, *Extreme Programming*) es un enfoque para el desarrollo de software que utiliza buenas prácticas de desarrollo y las lleva a los extremos. Se basa en valores, principios y prácticas esenciales. Los cuatros valores son la comunicación, la simplicidad, la retroalimentación y la valentía. Recomendamos a los analistas de sistemas que adopten estos valores en todos los proyectos que emprendan, no sólo cuando recurran a medidas de programación extrema.

Durante la fase de terminación de un proyecto, con frecuencia es necesario realizar ajustes en la administración del mismo. En el capítulo 3 veremos que XP puede garantizar la terminación exitosa de un proyecto ajustando recursos importantes como el tiempo, el costo, la calidad y el alcance. Cuando estas cuatro variables de control se incluyen adecuadamente en la planeación, se propicia un equilibrio entre los recursos y las actividades requeridas para completar el proyecto.