

SISTEMAS COMO CAMBIO ORGANIZACIONAL PLANEADO (13)

La construcción de un nuevo sistema de información es una forma de cambio organizacional planeado. Cuando se diseña un nuevo sistema de información, se está rediseñando la organización.

Desarrollo de sistemas y cambio organizacional

Existen cuatro tipos de cambio organizacional estructural que la tecnología de información hace posible:

1. Automatización: destinadas a ayudar a los empleados a realizar sus tareas con mayor eficiencia y eficacia. Ej: cálculos de sueldos y registros de nómina.

2. Estructuración: con frecuencia la automatización pone en evidencia cuellos de botella en la producción. La estructuración de procedimientos es la agilización de procedimientos operativos estandarizados.

La automatización y la estructuración son cambios lentos y presentan rendimientos modestos con poco riesgo.

3. Reingeniería de procesos de negocios: se analizan, simplifican y rediseñan los procesos de negocios. La reingeniería reorganiza los flujos de trabajo, combinando los pasos para reducir las pérdidas y eliminando las tareas repetitivas, de uso intensivo de papel. Este requiere una nueva visión de cómo debe organizarse el proceso.

Tanto la estructuración de procedimientos y el rediseño de los procesos de negocios están limitados a partes específicas de una empresa.

4. Cambio de paradigma: implica replantear la naturaleza del negocio, definir un nuevo modelo de negocios y con frecuencia cambiar la naturaleza misma de la organización. Ej: IBM paso de producción-venta de soft y hard a la consultoría informática.

Los cambios de paradigmas y la reingeniería suelen fallar porque es muy difícil dirigir el cambio organizacional generalizado. Sin embargo, conlleva recompensas altas en cuanto al rendimiento de sus inversiones (o productividad).

Reingeniería de procesos de negocios

Si las organizaciones replantean y rediseñan sus procesos de negocios antes de construir sistemas de información, pueden obtener rendimientos significativos de sus inversiones en tecnología de información.

El proceso de agilización de los procedimientos de negocios para que los documentos se puedan mover fácil y eficazmente se llama *administración del flujo de trabajo*.

El software de administración del flujo de trabajo y documentos automatiza los procesos como la distribución de los documentos a diferentes ubicaciones, la protección de aprobaciones, y la programación y generación de informes. Dos o más personas pueden trabajar simultáneamente en el mismo documento, lo cual permite lograr un tiempo de terminación más rápido. Y con un sistema de indexación diseñado adecuadamente, los usuarios podrán recuperar los archivos de diversas maneras, con base en el contenido del documento.

Pasos para lograr una reingeniería efectiva

Por lo general, la dirección general de la empresa es la que determina cuáles procesos de negocios requieren mejorarse y cuáles tienen prioridad más alta. Esta decisión resulta de considerar dos situaciones:

- **Análisis estratégico:** los directivos identifican los procesos de negocios que son más importantes para el éxito de la empresa y concentran esfuerzos en estos procesos.

- **Puntos de aflicción:** los directivos identifican los procesos que producen la mayor parte de las quejas de fabricantes, clientes o empleados y se aplican a arreglarlos primero.

Una vez que se han elegido los procesos de negocios apropiados, las empresas se ocupan de las siguientes actividades:

- ♣ **Identificar las entradas y salidas de un proceso de negocios:** las entradas se refieren a los elementos tangibles e intangibles que fluyen desde el entorno hacia el proceso. Las salidas fluyen de regreso al entorno.

- ♣ **Identificar el flujo de productos y/o servicios.**

- ♣ **Identificar la red de actividades y las holguras en el proceso.** Los procesos son conjuntos de actividades de negocios relacionados. Las holguras son tiempos de retardo integrados en el proceso para facilitar la revisión y vigilancia.

- ♣ **Identificar todos los recursos.** Todos los procesos consumen capital, mano de obra y tiempo. Es imperativo que se establezcan los costos iniciales. Éstos se refieren al costo del proceso de negocios original en cuanto a tiempo y recursos antes de que se realice cualquier mejora.

- ♣ **Identificar la estructura y el flujo de información.**

- ♣ **Identificar a los propietarios del proceso.** Los propietarios son aquellos encargados de tomar las decisiones y tienen la responsabilidad principal del proceso y las decisiones.

- ♣ **Identificar a los actores de los procesos y a los encargados de la toma de decisiones.**

Una vez que las empresas identifican y describen el proceso existente, el siguiente paso es comprender cuánto cuesta el proceso y el tiempo que se requiere para realizarlo.

- Costo del proceso de negocios para una transacción típica.
- El tiempo del proceso: el tiempo total de decisiones y actividades de todos los actores
- Calidad del proceso: la cantidad de tiempo y dinero gastado en volver a hacer partes y servicios defectuosos.
- Flexibilidad del proceso: la capacidad del proceso para producir una variedad de resultados, o de cambiar ante la presión del entorno.

Luego es pensar cómo mejoramos el proceso. Los siguientes son algunos de los principios que utilizan los diseñadores de procesos de negocios:

- ♣ Reemplazar los pasos secuenciales del proceso con pasos paralelos.
- ♣ Enriquecer los puestos de trabajo al incrementar su autoridad para tomar decisiones y concentrar la información.
- ♣ Permitir la compartición de la información entre todos los participantes
- ♣ Eliminar las holguras (retrasos en la toma de decisiones e inventarios)
- ♣ Transformar el procesamiento por lotes y la toma de decisiones en procesos de flujo continuo
- ♣ Automatizar las tareas de toma de decisiones donde sea posible.

James Champy, llama X-ingeniería al rediseño conjunto de los procesos de negocios interorganizacionales y es incluso más difícil implementarlo con éxito que los procesos de reingeniería para una sola compañía.

MEJORA DE PROCESOS

La administración de procesos de negocios y los programas de mejora de la calidad ofrecen oportunidades para realizar tipos de cambios de procesos de negocios más incrementales y continuos.

Administración de procesos de negocios (BPM)

BPM es un esfuerzo para ayudar a las empresas a manejar los cambios en los procesos como elementos fundamentales para los sistemas de información corporativos.

Incluye administración del flujo de trabajo, notación de modelado de los procesos de negocios, medición y administración de la calidad, administración del cambio y herramientas para modelar los procesos de negocios y utilizar los procesos de negocios de la empresa a una forma estandarizada, donde se puedan manipular continuamente.

Se utilizan herramientas de mapeo de procesos para identificar y documentar los procesos existentes y para crear modelos de procesos mejorados que después se puedan traducir en sistemas de software.

Un proceso de negocios se puede expresar totalmente en software y realizarse con poca o ninguna intervención de las personas.

La BPM también incluye vigilancia y análisis de procesos.

Administración de la calidad total (TQM)

TQM sostiene que el logro del control de la calidad es un fin en sí mismo. Se espera que todos contribuyan al perfeccionamiento total de la calidad: el ingeniero que evita los errores de diseño, el obrero de producción que localiza los defectos, el representante de ventas que presenta el producto adecuadamente a los clientes potenciales, e incluso la secretaria que evita los errores de mecanografía.

Seis sigma

Es una medida específica de calidad, que representa 3,4 defectos de partes por millón. Se utiliza seis sigma como una meta para implementar un conjunto de metodologías y técnicas para mejorar la calidad y reducir los costos. Las mejoras de calidad no sólo pueden elevar el nivel del producto y la calidad del servicio sino que también pueden bajar los costos.

Cómo apoyan los sistemas de información las mejoras de calidad

TQM se enfoca en hacer una serie de mejoras continuas en lugar de las drásticas explosiones de cambio.

Seis sigma usa las herramientas del análisis estadístico para descubrir las fallas en la ejecución de un procedimiento existente y hacer ajustes menores.

Los sistemas de información pueden ayudar a las empresas a lograr sus metas de calidad ayudándoles a simplificar productos o procesos, hacer mejoras con base en las demandas del cliente, reducir el tiempo de ciclo, aumentar la calidad y precisión del diseño y la producción, y a cumplir con estándares de benchmarking.

El **benchmarking** consiste en establecer estándares estrictos para productos, servicios y otras actividades, y después medir el desempeño contra estos estándares. Se pueden usar estándares de industrias externas, normas establecidas por otras compañías, altos estándares desarrollados internamente o alguna combinación de los tres.

VISIÓN GLOBAL DEL DESARROLLO DE SISTEMAS

Un nuevo sistema de información se construye como una solución para uno o varios problemas a que se esté enfrentando la organización.

Las actividades involucradas en la generación de una solución de sistemas de información para un problema organizacional o una oportunidad, se denominan **desarrollo de sistemas**. Éste consiste en una solución estructurada para un problema mediante distintas actividades, como

son: el análisis, diseño de sistemas, programación, pruebas, conversión, producción y mantenimiento.

1. Análisis de sistemas: es el análisis de un problema que una organización tratará de resolver con su sistema de información. Consiste en definir el problema, identificar sus causas, especificar la solución e identificar los requerimientos de información que debe satisfacer una solución de sistemas.

El analista elabora un panorama general de la estructura organizacional y los sistemas existentes, identificando a los principales responsables y usuarios de datos junto con el hardware y software existentes. Es decir, se detalla los problemas de los sistemas existentes. Mediante el examen de documentos de trabajo y procedimientos, la observación de operaciones de sistemas y la entrevista a usuarios clave de los sistemas, el analista puede identificar las áreas con problemas y los objetivos que tendrán que cumplir una solución.

El analista podría incluir un *estudio de factibilidad* para determinar si esa solución será factible, desde un punto de vista financiero, técnico u organizacional.

Establecimiento de requerimientos de información: esto implica identificar quién necesita qué información, dónde, cuándo y cómo. Este análisis define los objetivos del sistema nuevo o modificado y produce una descripción detallada de las funciones que el nuevo sistema debe realizar.

Algunos problemas no requieren una solución de sistemas de información sino tan sólo un ajuste administrativo, capacitacional adicional o un refinamiento de los procedimientos organizacionales existentes.

2. Diseño de sistemas: el diseño de un sistema de información es el modelo o plan general para ese sistema. Este consta de todas las especificaciones para dar forma y estructura al sistema.

Estas especificaciones deben abarcar todos los componentes administrativos, organizacionales y tecnológicos de la solución de sistemas.

Lo que hace que un diseño sea superior a otros es la facilidad o eficacia con que cumple con los requerimientos del usuario dentro de un conjunto específico de restricciones técnicas, organizacionales, financieras y de tiempo.

El rol de los usuarios finales: Los usuarios deben tener suficiente control sobre el proceso de diseño para asegurarse de que el sistema refleje sus prioridades de negocios y necesidades de información. Trabajar en el diseño aumenta la comprensión y aceptación del sistema por parte de los usuarios.

3. Programación: durante esta etapa, las características técnicas del sistema que se prepararon durante la etapa de diseño se traducen en código de programa.

4. Pruebas: Se deben realizar pruebas exhaustivas y completas para determinar si el sistema produce los resultados correctos.

Las pruebas consumen gran cantidad de tiempo: los datos de las pruebas se deben preparar cuidadosamente, repasar los resultados y hacer las correcciones en el sistema. En algunos casos se tiene que rediseñar parte del sistema.

Las pruebas pueden dividirse en tres etapas:

♣ *Prueba unitaria:* consiste en comprobar por separado cada programa del sistema, donde el propósito es garantizar que los programas están libres de errores. Los problemas se pueden corregir una vez localizados.

♣ *Prueba del sistema:* comprueba el funcionamiento en conjunto del sistema de información. Determina si los módulos independientes funcionarán juntos como se planeó y si realmente hay diferencias entre la manera en que el sistema trabaja y la manera en que fue concebido.

♣ *Prueba de aceptación:* proporciona la certificación final de que el sistema está listo para usarse en una situación de producción. Las pruebas de los sistemas son evaluadas por los

usuarios y revisadas por la administración. Cuando todas las partes están satisfechas de que el nuevo sistema cumple sus normas, se acepta formalmente la instalación del sistema

El **plan de pruebas** incluye todas las preparaciones para la serie de pruebas descritas.

5. Conversión: es el proceso de cambiar del sistema antiguo al nuevo. Pueden emplearse:

♣ *Una estrategia en paralelo:* el sistema anterior y su reemplazo potencial se ejecutan juntos durante un tiempo hasta que todos en la organización estén seguros de que el nuevo funcione correctamente. Sin embargo, este método es muy caro y puede requerir personal o recursos adicionales para ejecutar el sistema adicional.

♣ *Una estrategia de conversión directa:* reemplaza totalmente el sistema anterior con el nuevo en un día designado. Es muy arriesgado, donde las desarticulaciones, las interrupciones y el costo de las correcciones pueden ser enormes.

♣ *Una estrategia de estudio piloto:* presenta el nuevo sistema a solo un área limitada de la organización, como un solo departamento. Cuando esa versión piloto esté completa y trabajando sin problemas, se instala en toda la organización, ya sea simultáneamente o por etapas.

♣ *Una estrategia de enfoque por fases:* introduce el nuevo sistema en etapas, ya sea por funciones o por unidades organizacionales.

El cambio de un sistema antiguo por uno nuevo requiere que se capacite a los usuarios finales para que puedan usar el nuevo sistema. La documentación detallada que muestra cómo trabaja el sistema desde el punto de vista tanto técnico como del usuario final, se termina durante el tiempo de conversión con el propósito de utilizarla en la capacitación y las operaciones cotidianas.

6. Producción y mantenimiento: Una vez instalado el nuevo sistema y completada la conversión, se dice que el sistema está en producción. Los usuarios y los especialistas técnicos revisarán el sistema para determinar qué tan bien ha cumplido con sus objetivos originales y para decidir si se requiere alguna revisión o modificación. En algunos casos se preparará un documento formal de auditoría posterior a la implementación.

El **mantenimiento** son los cambios en el hardware, software, documentación o procedimientos para corregir errores en un sistema en producción, cumplir con los nuevos requerimientos o mejorar la eficacia del proceso.

MODELADO Y DISEÑO DE SISTEMAS

Metodologías estructuradas

Estructurada se refiere a que las técnicas se aplican paso a paso, y cada paso se construye con base en el anterior. Las metodologías estructuradas siguen un enfoque descendente, avanzando desde el nivel más alto y abstracto hacia el nivel más bajo de detalle.

Los métodos de desarrollo estructurados se orientan a procesos o acciones que capturan, almacenan, manipulan y distribuyen los datos a medida que fluyen a través de un sistema. Estos métodos separan los datos de los procesos.

Se debe escribir un procedimiento de programación separado cada vez que alguien necesita realizar una acción sobre una pieza de datos específica.

El DFD es la principal herramienta para representar los procesos de los componentes de un sistema y el flujo de datos que se origina entre ellos. Este diagrama especifica de manera precisa los procesos o transformaciones que ocurren dentro de cada módulo y las interfaces que existen entre ellos.

Los diagramas se pueden usar para delinear detalles de los procesos de niveles superiores así como los de niveles más bajos. A través de los diagramas de flujo de datos nivelados, un proceso complejo se puede desglosar en niveles sucesivos de detalle. Todo un sistema se puede dividir en un subsistema con un diagrama de flujo de datos de alto nivel. Cada subsistema a su vez, se puede dividir en subsistemas adicionales con DFD de segundo nivel, y los subsistemas de nivel menor se pueden desglosar de nuevo hasta el nivel más bajo de detalle que se haya alcanzado.

Otras herramientas es el diccionario de datos que contiene la información sobre las piezas individuales de datos y agrupaciones de datos dentro de un sistema. Este define los volúmenes de flujos y almacenes de datos para que los constructores de sistemas entiendan qué piezas de datos contienen. Las ***especificaciones de procesos*** describen la transformación que ocurre dentro del nivel más bajo de los DFD. Expresan la lógica de cada proceso.

El diseño de software se modela por medio de gráficas de estructura jerárquicas. La gráfica de estructura es un diagrama de arriba hacia abajo, que muestra cada nivel del diseño, su relación con los demás niveles y su lugar en la estructura total del diseño. El diseño toma en cuenta primero la función principal de un programa o sistema, después divide esta función en subfunciones y desglosa cada subfunción hasta el nivel más bajo de detalle que se haya alcanzado. Una gráfica de estructura puede documentar un programa, un sistema o parte de un programa.

Desarrollo orientado a objetos

Considera el objeto como la unidad básica del análisis y diseño de sistemas. Un objeto combina datos y los procesos específicos que operan sobre estos datos. Los datos encapsulados en un objeto sólo se pueden acceder y modificar mediante las operaciones o métodos, asociados con ese objeto. Los programas envían un mensaje para que un objeto realice una operación que ya está incorporada en él. El sistema se modela como una colección de objetos y de las relaciones que existen entre ellos. Los objetos deben colaborar entre sí para hacer que el sistema funcione.

El modelado orientado a objetos se basa en los conceptos de clase y herencia. Los objetos que pertenecen a cierta clase tienen las características de esa clase. Las clases de objetos pueden heredar toda la estructura y comportamientos de una clase más general y luego agregar variables y comportamientos únicos a cada objeto. Para crear nuevas clases de objetos se elige una clase existente y se especifica cómo se diferencia la nueva clase de la existente.

El desarrollo orientado a objetos es más iterativo e incremental que el desarrollo estructurado tradicional. Durante el análisis, los constructores de sistemas documentan los requerimientos funcionales del sistema, especificando sus propiedades más importantes y lo que el sistema propuesto debe hacer. Se analizan las interacciones entre el sistema y sus usuarios para identificar objetos, los cuales incluyen tanto datos como procesos. La etapa del diseño orientado a objetos describe cómo se comportarán los objetos y cómo interactuarán unos con otros. Los objetos similares se agrupan para formar una clase y las clases se agrupan en jerarquías, en las que una subclase hereda los atributos y métodos de su superclase.

El sistema de información se implementa traduciendo el diseño a código de programa, reutilizando clases que ya están disponibles en una biblioteca de objetos de software reutilizables y agregando las clases creadas durante la etapa del diseño orientado a objetos. La implementación también puede implicar la creación de una base de datos orientada a objetos. El sistema resultante debe probarse y evaluarse completamente.

Los nuevos sistemas se podrían crear usando algunos objetos existentes, cambiando otros y agregando nuevos objetos. Se han desarrollado marcos de trabajo orientado a objetos para proporcionar aplicaciones reutilizables, semicompletas, que la organización puede personalizar hasta obtener aplicaciones terminadas.

Ingeniería de software asistida por computadoras (CASE)

Es la automatización de metodologías paso a paso para el desarrollo de software y de sistemas para reducir la cantidad de trabajo repetitivo que el diseñador necesita hacer. Se facilita la creación de documentación estructurada y la coordinación de los esfuerzos de desarrollo del equipo. Los miembros del equipo pueden compartir su trabajo accediendo a los archivos de cada uno para revisar o modificar lo que se ha hecho. Muchas herramientas CASE se basan en Pc, con capacidades gráficas robustas.

Las herramientas CASE proporcionan medios gráficos automatizados para producir gráficas y diagramas, generadores de pantallas e informes, diccionarios de datos, medios de informes extensos, herramientas de análisis y verificación, generadores de códigos y generadores de documentación. Las herramientas CASE tratan de aumentar la productividad y la calidad al:

- Aplicar una metodología de desarrollo estándar e imponen disciplina en el diseño
- Mejoran la comunicación entre los usuarios y los especialistas técnicos
- Organizan y correlacionan los componentes de diseño, y proporcionan acceso rápido a ellos a través de un repositorio de diseño
- Automatizan las partes tediosas y propensas a error del análisis y el diseño
- Automatizan la generación de código y prueban y controlan la implementación

Las herramientas CASE para el **front end** se enfocan en la captura de información para el análisis y el diseño en las primeras etapas de desarrollo de sistemas. CASE para el **back end** determinan las actividades de codificación, prueba y mantenimiento. Estas ayudan a convertir automáticamente las especificaciones en código de programa.

Las herramientas CASE enlazan automáticamente los elementos de datos con los procesos en que se usan. También contienen características para la validación de diagramas y especificaciones de diseño. Por lo tanto, apoyan el diseño iterativo automatizando las revisiones y los cambios así como proporcionando los medios para la creación de prototipos. Un repositorio de información CASE almacena toda la información definida por los analistas durante el proyecto. Este incluye DFD, gráficas de estructura, diagrama de entidad relación, definiciones de datos, especificaciones de procesos, formatos de pantalla e informes, notas, comentarios y resultados de la prueba.

Para usarla eficazmente requieren disciplina organizacional. Todo miembro de un proyecto de desarrollo debe aceptar un conjunto común de convenciones de nombres y estándares, así como una metodología de desarrollo.

MÉTODOS ALTERNATIVOS PARA LA CONSTRUCCIÓN DE SISTEMAS

Los sistemas difieren en cuanto a tamaño, complejidad tecnológica y problemas organizacionales que están destinados a resolver. Se han desarrollado diversos métodos para la construcción de sistemas destinados a ocuparse de estas diferencias.

El ciclo de vida tradicional de los sistemas

Es un método tradicional para desarrollar un sistema de información, que divide el proceso de desarrollo en etapas formales que se deben llevar a cabo en secuencia. Hay una división muy formal del trabajo entre los usuarios finales y los especialistas en sistemas. Los especialistas técnicos, son responsables de gran parte del trabajo de análisis, diseño e implementación de sistemas; los usuarios finales están limitados a proporcionar los requerimientos de información y a revisar el trabajo del personal técnico. También pone énfasis en las especificaciones formales y el trabajo administrativo, por lo que se generan demasiados documentos durante el curso de un proyecto de sistemas.

Puede ser costoso, lento e inflexible. Éste es un método de “cascada” en el que las tareas de una etapa se completan antes de que empiece el trabajo de la siguiente etapa. Las actividades se

pueden repetir, pero se deben generar volúmenes de nuevos documentos y volver a seguir los pasos si es necesario revisar los requerimientos y las características técnicas. Este método no es conveniente para muchos sistemas pequeños que son menos estructurados y más individualizados.

El ciclo de vida tradicional se sigue usando para construir grandes sistemas, como TPS o MIS, donde las necesidades están muy estructuradas y bien definidas. Este enfoque es costoso, tardado e inflexible lo que obstaculiza cambios. Es necesario generar gran cantidad de documentos, y no es apropiado para aplicaciones orientadas a la toma de decisiones.

Creación de Prototipos

La elaboración de un prototipo es un proceso rápido y económico a fin de que los usuarios puedan determinar mejor sus necesidades de información. Una vez en operación se va retocando hasta que cumpla con los requisitos del usuario. Es iterativo porque los pasos se repiten una y otra vez.

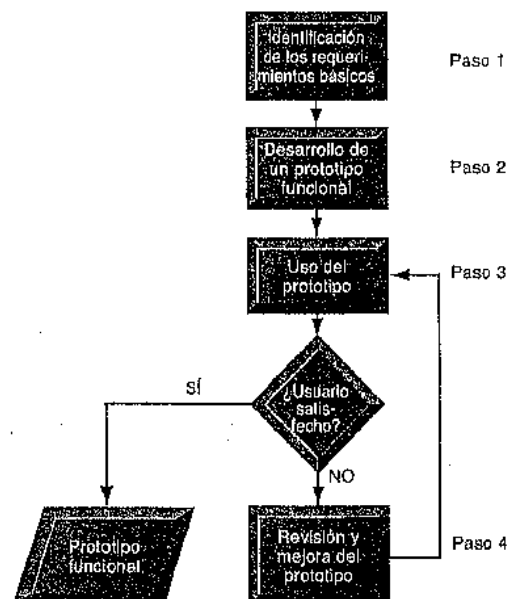
Etapas:

1. Identificar los requerimientos básicos del usuario
2. Crear un prototipo inicial.
3. Usar el prototipo.
4. Revisar y mejorar el prototipo. Se toma nota de todos los cambios que el usuario solicita y a continuación mejora el prototipo

Los pasos 3 y 4 se repiten hasta que el usuario está satisfecho. Cuando ya no se requiere ninguna iteración, el prototipo aceptado se vuelve un prototipo funcional que proporciona las últimas especificaciones para la aplicación.

El prototipo es una versión funcional de un sistema de información, su propósito es servir de modelo preliminar. Una vez finalizado el diseño, el prototipo se puede convertir en un sistema de producción refinado.

Ventajas: son más útiles cuando existe cierta incertidumbre acerca de las necesidades o de las soluciones de diseño, y son más apropiados para aplicaciones pequeñas. Además fomenta la participación intensa de los usuarios finales en todas la etapas. El uso de prototipos es valioso para diseñar interfases de usuarios de un sistemas, ya que es la parte con la que los usuarios interactúan.



Desventaja: si el prototipo terminado funciona razonablemente, se podría caer en el error de considerar que las tareas de volver a programar o diseñar, de documentar y probar todo el sistema, son innecesarias para construir un sistema de producción refinado.

Desarrollo por usuarios finales

Los propios usuarios, con poca o ninguna ayuda de especialistas, desarrollan sistemas de información. Se utilizan los lenguajes de cuarta generación, lenguajes gráficos y herramientas de software para tales fines.

Los lenguajes de cuarta generación consisten en herramientas de software que habilitan a los usuarios finales para crear informes o desarrollar aplicaciones de software con un mínimo de asistencia técnica. Estos tienden a ser no procedurales. Los lenguajes procedurales requieren una especificación de secuencia de pasos, que le indican a la computadora qué hacer y cómo hacerlo. Los lenguajes no procedurales sólo necesitan especificar qué se tiene que efectuar.

Los lenguajes de consultas son herramientas de software que proporcionan respuestas inmediatas en línea a solicitudes de información no predefinidas como: ¿Quiénes son los representantes de ventas de más alto desempeño?

Ventajas: Los sistemas de esta clase se crean con gran rapidez. Otras ventajas que presentan son la determinación más exacta de necesidades, mayor participación y satisfacción del usuario y reducción del número de aplicaciones pendientes de implementar, ya que no dependen de los especialistas. Sin embargo, las herramientas de cuarta generación todavía no pueden reemplazar a las herramientas convencionales para algunas aplicaciones de negocios porque no pueden manejar fácilmente el procesamiento de volúmenes grandes de transacciones o aplicaciones que utilizan mucha lógica de procedimientos y requieren actualizaciones constantes.

Desventajas: cuando se crean los sistemas rápidamente, sin una metodología de desarrollo formal, la prueba y la documentación podrían no ser adecuadas. Para ayudar a las organizaciones a maximizar los beneficios del desarrollo de aplicaciones por parte del usuario final, la administración debe controlar el desarrollo de aplicaciones por parte del usuario final requiriendo la justificación del costo de los proyectos de sistemas de información del usuario final y estableciendo hardware, software y normas de calidad para las aplicaciones desarrolladas por el usuario.

Paquetes de software de aplicación

Es un conjunto de programas de aplicación preescritos y precodificados, que se encuentran en el mercado para su venta o su arrendamiento. Hay muchas aplicaciones que efectúan tareas comunes para todas las organizaciones de negocios. Si se puede adquirir un paquete apropiado en muchos casos no es necesario que la compañía desarrolle sus programas.

Los programas del paquete se probaron antes de salir al mercado, por lo que la prueba del comprador se realizan en breve tiempo. Los proveedores ofrecen buena parte del mantenimiento y ofrecen mejoras de acuerdo a los adelantos técnicos y de negocios.

Las funciones de **personalización** permiten modificar un paquete de modo que satisfaga las necesidades singulares de una organización, sin destruir la integridad del mismo. Pero esta personalización podría llegar a ser muy costosa y consumir demasiado tiempo, por lo que se podrían perder las ventajas del paquete de software de aplicación.

Cuando se desarrolla un sistema utilizando un paquete de software de aplicaciones, el análisis de sistemas incluirá un proyecto de evaluación del paquete. Este se basa en una solicitud de propuesta. La **solicitud de propuesta** es una lista de preguntas detalladas que se presentan a los proveedores de software, para determinar que tan bien el producto puede satisfacer las necesidades específicas de la organización.

Si los requerimientos de la organización no son compatibles con la manera de funcionar del paquete y éste no puede personalizarse, la organización tendrá que adaptarse al paquete y cambiar sus procedimientos.

Subcontratación

Los proveedores de servicios de aplicaciones son una forma de subcontratación. Las compañías suscriptoras podrían usar el software y el hardware de cómputo proporcionados por el proveedor como plataforma técnica para sus sistemas.

Otra forma, una compañía podría contratar a un proveedor externo para que diseñe y elabore el software para su sistema, pero esa compañía operaría el sistema en sus computadoras propias. El proveedor de la subcontratación podría ser nacional o encontrarse en otro país.

El proveedor de servicios de subcontratación se beneficia de las economías de escala y de competencias centrales complementarias que serían difíciles de duplicar para una empresa que no se especializa en servicios de tecnología de información.

La subcontratación permite que una compañía pague sólo por lo que usa en lugar de construir su propio centro de computación, que se subutilizaría cuando no hubiera una carga máxima.

Muchos subestiman los costos de identificar y evaluar a fabricantes de servicios de tecnología de información, hacer la transición a un nuevo fabricante y monitorear a los fabricantes para asegurarse de que estén cumpliendo con sus obligaciones contractuales. Estos costos ocultos pueden reducir fácilmente los beneficios provistos de la subcontratación.

Cuando una empresa asigna la responsabilidad de desarrollar y operar sus sistemas de información a otra organización, es posible que pierda el control sobre su función de sistemas de información. Si la organización no tiene habilidad para negociar un contrato favorable, la dependencia que la empresa tendría del fabricante podría producir altos costos o pérdida de control sobre la dirección tecnológica.

Es más probable que una empresa obtenga beneficios de la subcontratación si entiende sus propios requerimientos, sabe exactamente la manera en que el fabricante subcontratista le proporcionará valor, e identifica a los proveedores con capacidades y objetivos que mejor se alineen con sus necesidades específicas. La empresa también debe ser capaz de diseñar un contrato para la subcontratación que le permita manejar la relación con el fabricante.

Desarrollo rápido de aplicaciones (RAD)

Las herramientas del software orientados a objetos, el software reutilizable, la creación de prototipos y las herramientas de cuarta generación están ayudando a los constructores de sistemas a crear sistemas funcionales mucho más rápido.

El término desarrollo rápido de aplicaciones (RAD) se usa para describir este proceso de crear sistemas funcionales en un período muy corto. El RAD puede incluir el uso de la programación visual y otras herramientas para construir interfaces gráficas de usuario, la creación de prototipos iterativos de elementos clave del sistema, la automatización de generación de código de programa en intenso trabajo en equipo entre usuarios finales y especialistas de sistemas de información.

Se pueden ensamblar sistemas sencillos a partir de los componentes prefabricados. El proceso no tiene que ser secuencial y las partes clave del desarrollo se pueden realizar de manera simultánea.

Hay ocasiones en las que se usa una técnica llamada **diseño conjunto de aplicaciones (JAD)**, para acelerar la generación de requerimientos de información y para desarrollar el diseño inicial de sistemas. El JAD junta a los usuarios finales y especialistas en sistemas en una sesión interactiva para discutir el diseño del sistema. Las sesiones JAD pueden acelerar significativamente la etapa del diseño e involucrar a los usuarios en un nivel intenso.

Desarrollo basado en componentes y servicios web

El desarrollo basado en componentes permite construir un sistema ensamblando e integrando componentes de software existentes.

Servicios web y computación orientada a servicios: Los servicios web se pueden utilizar como herramientas para construir nuevas aplicaciones de sistemas de información o mejorar los sistemas existentes. Los servicios web pueden crear componentes de software que se distribuyen a través de internet y proporcionan nuevas funciones para los sistemas existentes de una organización o pueden crear nuevos sistemas que enlacen los sistemas de una organización con los de otras organizaciones. Como estos servicios de software utilizan un conjunto universal de estándares, prometen ser menos costosos y menos difíciles de integrar que los componentes propietarios.

Los servicios web pueden realizar ciertas funciones por sí mismos, y también pueden involucrar a otros servicios Web para completar transacciones más complejas. Al crear componentes de software que puedan comunicar y compartir datos sin que importe el sistema operativo, lenguaje de programación o dispositivo del cliente, los servicios Web pueden proporcionar ahorros significativos de costos en la construcción de sistemas al mismo tiempo que abren nuevas oportunidades para la colaboración con otras empresas.