

Los Sistemas Gestores de Flujos de Trabajo en la Gestión de Procesos Software

The Workflow Management Systems for Software Process Management

Víctor Hugo Menéndez Domínguez¹
mdoming@correo.uady.mx

María Enriqueta Castellanos Bolaños¹
enriqueta.c@correo.uady.mx

¹ Universidad Autónoma de Yucatán, México

Resumen: Hoy en día, las empresas de software se ven obligadas a mejorar su rendimiento para poder sobrevivir en un entorno altamente competitivo: los clientes demandan una mayor calidad, menor plazo de entrega, mejor servicio, atención personalizada, etc. En este trabajo se describen a los procesos software así como los flujos de trabajo. Su combinación y automatización mejora el flujo de información asociado a las actividades de dichos procesos, lo que facilita el control y permite mejorar el rendimiento, reducir costos y plazos, minimizar errores y garantiza una mayor calidad, entre otras ventajas.

Palabras clave: Proceso software, flujo de trabajo, Sistemas Gestores de Flujos de Trabajo, Workflow Management System, WfMS

Abstract: Nowadays, software companies are forced to improve their performance in order to survive in a highly competitive environment: customers demand higher quality, shorter delivery, best service, personalized attention, etc. This paper describes the software processes and workflows. Their combination and automatization improve the flow of information associated with the activities of such processes, which facilitates control and improves performance, reduces costs and time, minimizes errors and ensures greater quality, among other benefits.

Keywords: Software process, workflows, WorkFlow Management System, WfMS

1. Introducción

Un proceso software se define como un marco de trabajo para las tareas que se requieren en la construcción de software de alta calidad (Pressman, 2014). Es decir, un proceso software es un conjunto de actividades y resultados asociados que producen un producto software (Sommerville, 2015).

El proceso software constituye la base para el control de la gestión de proyectos de software y establece el contexto en el cual se aplican los métodos, técnicas, se generan los productos del trabajo (modelos, documentos, datos, reportes, formatos, etc.) se establecen los fundamentos, se asegura la calidad, y el cambio se maneja de manera apropiada (Pressman, 2014; Berander y Wohlin, 2003).

Distintos procesos software organizan las actividades de diferentes formas y las describen con diferente nivel de detalle. El tiempo de cada actividad varía, así como los resultados. Aunque existen muchos procesos software diferentes, existen actividades fundamentales que son comunes para todos ellos. Los detalles (tareas) serán muy diferentes en cada caso, pero las actividades son iguales. La adaptación del proceso software es esencial para el éxito del proyecto. (Chan y Leung, 1997)

Para Sommerville (2015), estas actividades fundamentales son cuatro:

1. Especificación del software: debe definirse la funcionalidad del software y las restricciones sobre su operación.
2. Desarrollo del software: debe producirse software que cumpla con la especificación.
3. Validación del software: el software debe evaluarse para asegurar qué es lo que el cliente requiere.
4. Evolución del software: el software debe evolucionar para cumplir con los cambios requeridos por el cliente.

Por otro lado, Pressman (2014) establece estas actividades de forma distinta:

1. Comunicación: implica una intensa colaboración y comunicación con los clientes (stakeholders) además abarca la investigación de requisitos y otras actividades relacionadas.
2. Planeación: establece un plan para el trabajo de la ingeniería de software. Describe las tareas técnicas que deben realizarse, los riesgos probables, los recursos que serán requeridos, los productos del trabajo que han de producirse y un programa de trabajo.
3. Modelado: abarca la creación de modelos que permiten al desarrollador y al cliente entender mejor los requisitos del software y el diseño que logrará satisfacerlos.
4. Construcción: combina la generación del código (ya sea manual o automatizado) y la realización de pruebas necesarias para descubrir errores en el código.
5. Despliegue: el software (como entidad completa o un incremento completado de manera parcial) que se entrega al cliente, quién evalúa el producto recibido y proporciona información basada en su evaluación.

Además, define una serie de actividades paralelas que son aplicables a lo largo de todos los procesos software y se enfocan de modo principal en la gestión, el rastreo y el control del proyecto (Pressman, 2014):

1. Seguimiento y control del proyecto de software: permite que el equipo de software evalúe el progreso comparándolo con el plan de proyecto y así tomar las acciones necesarias para mantener el programa.
2. Gestión del riesgo: evalúa los riesgos que pudieran afectar los resultados del proyecto o la calidad del producto.
3. Aseguramiento de la calidad del software: define y conduce las actividades requeridas para asegurar la calidad del software.

4. Revisiones técnicas formales: evalúa los productos del trabajo de la ingeniería de software en un esfuerzo encaminado a descubrir y eliminar los errores antes de que estos se propaguen hacia la siguiente acción o actividad.
5. Medición: Define y recolecta mediciones del proceso, el proyecto y el producto para ayudar al equipo a entregar software que satisfaga las necesidades del cliente se puede usar en conjunto con todas las otras actividades del marco de trabajo o actividades sombrilla.
6. Gestión de la configuración del software: maneja los efectos del cambio a través del proceso del software.
7. Gestión de la reutilización: define los criterios para la reutilización de productos de trabajo (se incluyen componentes de software) y establece mecanismos para la creación de componentes reutilizables.
8. Preparación y producción del producto de trabajo: abarca las actividades requeridas para crear productos del trabajo como modelos, documentos, registros, formatos y listas.

Todos estos procesos software son descritos mediante modelos que se presentan desde una perspectiva particular. Por su naturaleza, los modelos son simplificaciones, por lo tanto un modelo de procesos software es una abstracción de un proceso real. Algunos ejemplos de estos tipos de modelos que se pueden producir son (Sommerville, 2015):

- Modelo de flujo de trabajo: muestra la secuencia de actividades en el proceso en conjunción con sus entradas, salidas y dependencias.
- Modelo de flujo de datos o de actividad: representa el proceso como un conjunto de actividades, cada una de las cuales lleva alguna transformación en los datos. Muestra cómo la entrada en el proceso, se transforma en una salida.
- Modelo de rol/acción: representan los roles de la gente involucrada en el proceso de software y las actividades de las que son responsables.

Los modelos genéricos de procesos deben adaptarse para que los utilice un equipo de proyecto de software. Para lograrlo se han desarrollado herramientas destinadas a ayudar a las organizaciones de software a analizar sus procesos actuales, organizar sus tareas en flujos de trabajo, controlar y monitorear su progreso, y administrar su calidad técnica.

Su origen está en la necesidad de mantener la integridad en los documentos generados como parte de un proceso en el que está involucrado un grupo de personas a lo largo de un período de tiempo.

2. Flujos de Trabajo

Un flujo de trabajo (o Workflow) es la automatización de un proceso de negocio (o de software), total o parcial, en la cual documentos, información o tareas son pasados de un participante a otro para efectos de su procesamiento, de acuerdo a un conjunto de reglas establecidas para obtener, o contribuir a los objetivos de la empresa (Hollingsworth, 1995).

Un Workflow involucra un número de pasos lógicos, cada uno conocido como una actividad. Una actividad puede involucrar interacción manual con un usuario o con otro flujo de trabajo o la actividad puede ser ejecutada utilizando algún recurso-máquina (Allen, 2001).

En otras palabras, un flujo de trabajo es, según Plesums (2002):

- Una secuencia de tareas donde intervienen participantes,
- Cada participante desempeña un papel (rol),
- Unos productos se manipulan y se transfieren entre participantes,
- Se hace uso de una información que fluye entre las tareas,
- De acuerdo con un conjunto de reglas y procedimientos,
- Para alcanzar un objetivo.

Los flujos de trabajo son un camino para la información, para reducir tiempo, dinero y esfuerzo en la ejecución de un proceso de negocio.

Mientras que los Workflow pueden ser gestionados manualmente, en la práctica están organizados en el contexto de un sistema informático que proporciona un soporte computarizado para una sistematización procedural (Hollingsworth, 1995; Pressman, 2014).

Dicha sistematización permite mejorar el rendimiento de los procesos, reducir los costos y los plazos, minimizar errores y garantizar una mayor calidad. Los sistemas que permiten automatizar los procesos de negocio reciben el nombre de Sistemas Gestores de Flujos de Trabajo y se apoyan en la utilización de las Tecnologías de la información.

3. Sistemas Gestores de Flujos de Trabajo

La gestión de flujos de trabajos para procesos requiere de herramientas software que incluyan elementos para definirlos, ejecutarlos, interfaces para utilizarlos, compartirlos, monitorearlos y controlarlos (Pressman, 2014).

Esta sección describe estas herramientas software llamadas Sistemas Gestores de Flujos de Trabajo (Workflow Management System, WfMS) y su relación con la gestión de procesos software.

La necesidad de reorganizar y mejorar los procesos de negocios y los avances en la tecnología de la información han creado un gran mercado para los Sistemas Gestores de Flujos de Trabajo (Casati, et al., 1996).

Un Sistema Gestor de Flujos de Trabajo (o simplemente sistema Workflow) proporciona una sistematización de los procesos de negocios mediante la gestión de la secuencia de actividades que los componen asignándolas a personas o procesos informáticos que las llevarán a cabo (Hollingsworth, 1995).

Los sistema Workflow definen, crean y gestionan la ejecución de flujos de trabajo mediante el uso de software (motores de ejecución) que son capaces de interpretar la definición del proceso, interactuar con los participantes y cuando es requerido ejecutar herramientas y aplicaciones (Berger et al., 1997).

Los Sistemas Gestores de Flujos de Trabajo ayudan a manejar los procesos de negocios asegurando que las tareas son priorizadas para ser llevadas a cabo (Plesums, 2002):

- Tan pronto como sea posible,
- Por la persona correcta,
- En el orden correcto.

Los sistemas Workflow se originaron como una forma de reducir el tiempo y costo de llevar a cabo los procesos de negocios y asegurar que las tareas sean llevadas a cabo consistentemente para mejorar la calidad. Se basan sobre la idea de que algunas cosas son realizadas más efectivamente por las computadoras que por las personas: búsqueda, control, etc. (Halliday, 2001)

Generalmente son utilizados por las empresas cuando es necesario sistematizar procesos de negocios que tienden a ser repetitivos, bien estructurados y con gran manejo de datos, o bien cuando se requiere mantener el control y la integridad sobre las modificaciones realizadas en un documento por un grupo de personas con el objetivo de lograr una meta común. Esto está relacionado con procesos administrativos relacionados con reportes de venta, ordenes de compras, control de inventarios, etc.

3.1 Características

Los Sistemas Gestores de Flujos de Trabajo permiten, diseñar, ejecutar y monitorear los flujos de trabajos en entornos distribuidos e incluso heterogéneos, permitiendo una gestión y ejecución eficiente de los procesos (Casati et al., 1996).

Una característica básica que debe proporcionar un sistema Workflow es la flexibilidad. En un entorno tan cambiante, las compañías necesitan refinar sus procesos constantemente con el propósito de cubrir las necesidades y oportunidades propuestas por nuevas tecnologías, nuevos requerimientos de mercado y nuevas leyes (Casati et al., 1996).

Las funciones más comunes que proporcionan los Sistemas Gestores de Flujos de Trabajo son (Plesums, 2002):

- Asignación de tareas al personal,
- Aviso al personal de tareas pendientes,
- Permitir la colaboración en las tareas comunes,
- Optimización de recursos humanos y técnicos, alineándolos a la estrategia de la empresa,
- Automatización de las secuencias de los procesos de negocio y optimización de las mismas,
- Agilización de los procesos de negocio y como resultado un mejor servicio al cliente,
- Control y seguimiento de dichos procesos,
- Interfaz para el acceso a sistemas de datos,
- Registro y monitoreo de acciones realizadas.

3.2 Ventajas

Los beneficios que se obtienen al emplear un Sistema Gestor de Flujos de Trabajo pueden ser divididos en tangibles e intangibles (Plesums, 2002; Prior, 2003):

Beneficios tangibles:

- Reducción en los costos de operación,
- Enfoque a la disminución de trámites,
- Mejora en la productividad,
- Disminución en los tiempos de procesamiento,
- Reducción en los tiempos de respuesta,
- Monitoreo de estado, recursos disponibles,
- Asignación de tareas de forma adecuada,
- Control de procesos,
- Integración con otras aplicaciones.

Beneficios intangibles:

- Mejora en los servicios,
- Mejora en las condiciones de los empleados,
- Mejora en la gestión del cambio,
- Calidad,
- Mejora en la comunicación,
- Soporte para la toma de decisiones,
- Mejora en la capacidad de planeación,
- Mejora en la capacidad de distribución,
- Comunicación entre organizaciones.

3.3 Requerimientos

Emplear un sistema Workflow para resolver una problemática en una empresa implica primeramente establecer de forma clara un modelo de procesos de negocios que incorpore los elementos que serán incluidos en el sistema:

- Tareas
- Personas (Usuarios),
- Roles,
- Reglas de transición,
- Datos,
- Eventos,
- Plazos (Deadlines),
- Procesos,
- Políticas.

Generalmente esta modelación se realiza utilizando un lenguaje de modelado de procesos como BPMN (OMG, 2006), el cual proporciona una notación que es fácilmente entendible para todos los usuarios relacionados con el sistema

(usuarios finales, analistas, desarrolladores y gestores de procesos). Su finalidad es proporcionar un puente entre el diseño del proceso de negocios y su implementación dentro de un sistema Workflow.

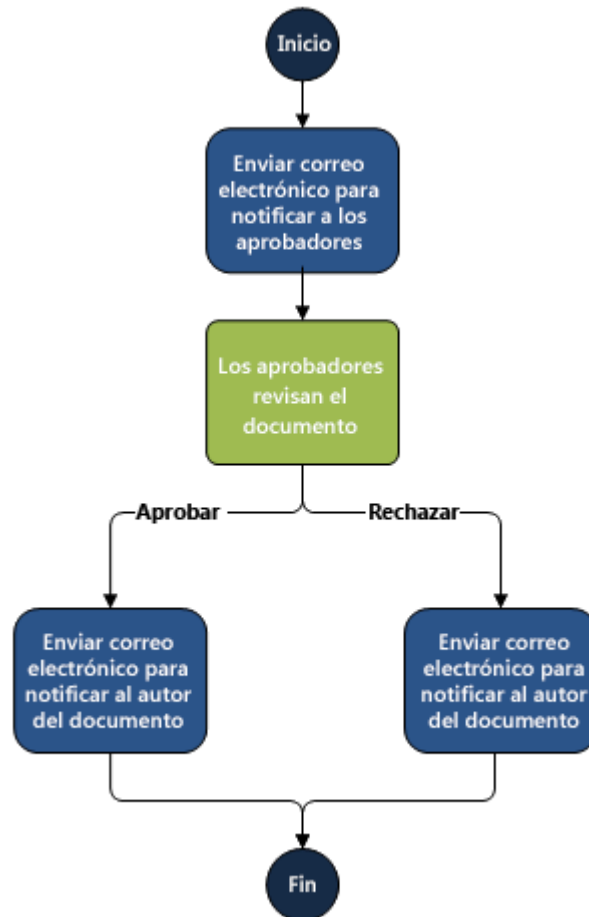


Figura 1. Ejemplo de un modelo de proceso de negocio (Microsoft, 2016).

Posteriormente está el emplear o desarrollar un sistema Workflow que se adapte a las necesidades de la empresa. En este sentido existen varias opciones en el mercado.

Por ejemplo, Microsoft Office SharePoint Server 2007 (Microsoft, 2016) proporciona herramientas que facilitan la colaboración en la edición de documentos y administrar tareas del proyecto tales como ver la lista de flujos de trabajo disponibles para un documento, iniciar un flujo de trabajo, reasignar una tarea del flujo de trabajo o completarla. Ofrece Windows Workflow Foundation,

una API que permite al programador crear flujos de trabajo con código personalizado que pueden utilizarse dentro del mismo entorno e integrarse con otros sistemas.

Por otra parte, Flokzu (<http://www.flokzu.com/ES>), intalio (<http://www.intalio.com/>) o jBPM (<http://www.jbpm.org/>) son Sistemas Gestores de Flujos de Trabajo que proporcionan una serie de características que las hacen una opción interesante:

- Trabajo en equipo,
- Crear modelos de procesos en formato BPMN,
- Diseñar formularios de captura,
- Almacenamiento en la nube,
- Integración con otros sistemas,
- Generación de estadísticas, entre otros.

3.4 Modelo de Referencia

Con el propósito de establecer un estándar para el desarrollo de Sistemas Gestores de Flujos de Trabajo, la Workflow Management Coalition (WfMC) ha establecido un modelo de referencia para Sistemas Gestores de Flujos de Trabajo (Hollingsworth, 1995) mostrado en la figura 2.

En el modelo adoptado hay una separación entre los procesos y el control de la lógica de las actividades. Esta separación permite la integración de las diversas herramientas con una aplicación particular. La interacción del servicio de ejecución con los recursos externos se da por una de las dos interfaces siguientes:

1. La interfaz de las Aplicaciones cliente, a través de la cual el Motor de Workflow interactúa con el manejador de la lista de trabajos (Worklist), responsable de organizar el trabajo por intermedio de un recurso de usuario. Es responsabilidad de este manejador hacer progresar cada elemento de la Worklist.

2. La interfaz de las Aplicaciones invocadas, la cual le permite al motor de Workflow activar una herramienta para realizar una actividad particular. Esta interfaz podría ser basada en un servidor, es decir no existe la interacción con el usuario.
3. Milk

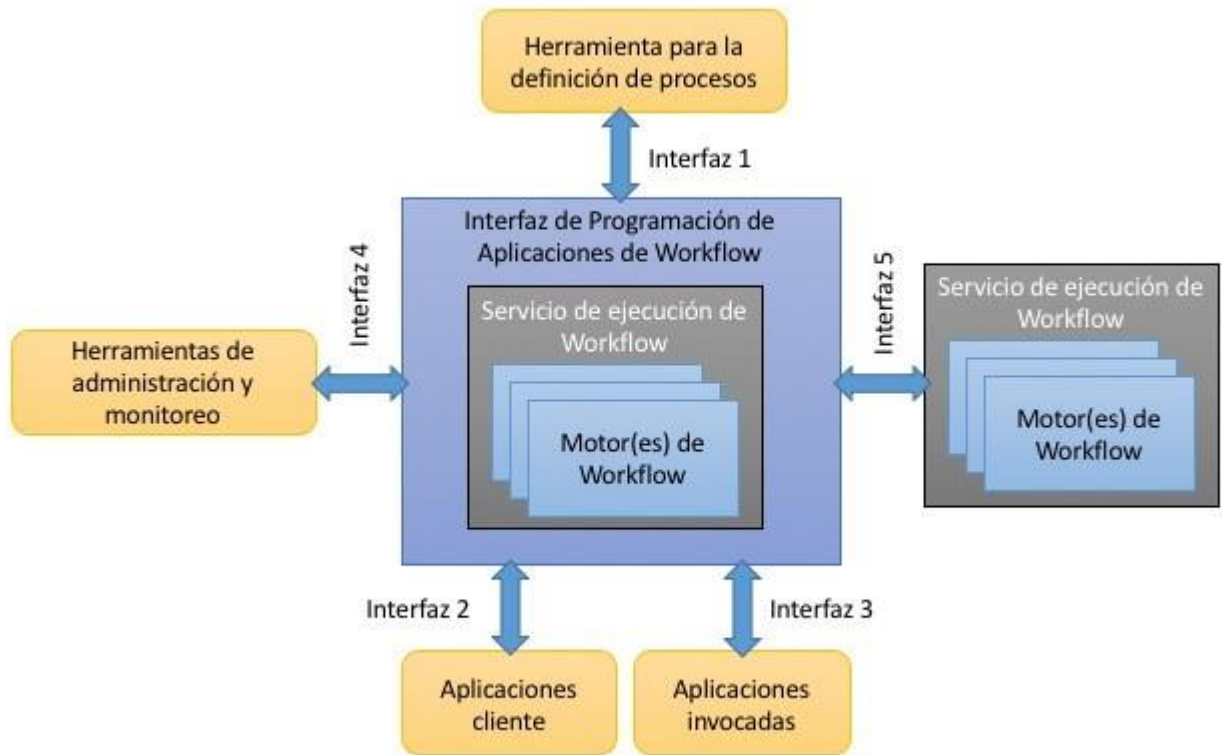


Figura 2. Modelo de Referencia WfMS.

A continuación se describen los principales componentes del modelo:

Motor de Workflow (Workflow Engine). El motor de Workflow es el software que provee el control del ambiente de ejecución de una instancia de flujo de trabajo. Típicamente el motor provee facilidades para:

- • Interpretación de la definición de procesos.
- • Control de las instancias de los procesos: creación, activación, terminación, etc.
- • Navegación entre actividades.
- • Soporte de interacción con el usuario.

- • Control de datos al usuario o hacia aplicaciones.
- • Invocación de aplicaciones externas.

Servicio de ejecución de Workflow (Workflow Enactment Service). Este componente interpreta la descripción de procesos y controla las diferentes instancias de los procesos, secuencia de actividades, adiciona elementos a la lista de trabajo de los usuarios, e invoca aplicaciones necesarias. Todas estas tareas son hechas por uno o más motores de Workflow, los cuales manejan la ejecución de las distintas instancias de varios procesos.

Interfaz de Programación de Aplicaciones de Workflow (WAPI). Las WAPI pueden ser vistas como un conjunto de API's (Application Programming Interface) y funciones de intercambio soportadas por el servicio de representación de Workflow. Las WAPI permiten la interacción del servicio de representación de Workflow con otros recursos y aplicaciones.

Herramientas de definición de procesos (interfaz 1). Existe una gran variedad de herramientas utilizadas para el análisis de procesos. Estas herramientas pueden variar desde las más informales hasta las más formales y sofisticadas. La salida de este proceso de modelado y diseño es una "definición de procesos" la cual puede ser interpretada en tiempo de ejecución por el o los motores de Workflow.

Aplicaciones clientes (interfaz 2). En el modelo planteado, la interacción entre las aplicaciones clientes y el motor de Workflow está sostenido en gran parte por el concepto de lista de trabajo ya descrito anteriormente. Parte de la información almacenada en la lista de trabajo es utilizada para transmitirle al manejador de la lista de trabajo qué aplicaciones hay que invocar. La interfaz entre una aplicación cliente de Workflow y el motor de Workflow debe ser lo suficientemente flexible en los siguientes puntos: identificadores de procesos y actividades, estructuras de datos, diferentes alternativas de comunicación.

Aplicaciones invocadas (interfaz 3). Esta interfaz está orientada a interactuar con agentes de una aplicación, o con toda la aplicación. Dichas aplicaciones deben estar orientadas al contexto general de un sistema de Workflow, es decir, deben poder interactuar directamente con el motor de Workflow. La aplicación invocada es manejada localmente por un motor de Workflow, usando la información suministrada en la definición del proceso para identificar la naturaleza de la actividad. La aplicación invocada puede ser local al motor de Workflow, es decir, residente en la misma plataforma, o estar en otra plataforma dentro de una red. En este caso la definición del proceso debe contener información necesaria para poder encontrar la aplicación que se va a invocar.

Funciones de interoperabilidad WAPI (interfaz 4). Existen dos aspectos necesarios para la interoperabilidad:

- Alcance de la interpretación común de la definición de procesos que será realizada.
- Soporte en tiempo de ejecución para el intercambio de diferentes tipos de información de control y transferencia de los datos relevantes del Workflow, y/o de las aplicaciones entre los distintos servicios de representación.

Herramientas de administración y monitoreo (interfaz 5). El propósito de esta interfaz es permitir una vista completa del estado del flujo de trabajo, además de poder realizar auditorías sobre los datos del sistema.

3.5 Clasificación

Debido a la diversidad de procesos de negocio que existen dentro de las empresas, es posible clasificar a los Sistemas Gestores de Flujos de Trabajo según su funcionalidad principal.

Para Plesums (2002), estos pueden ser clasificados en:

- **Producción.** Frecuentemente este tipo de sistemas Workflow es llamado Workflow de Transacciones. Esto se debe a que la transacción en una base de

datos es considerada la clave de todo proceso. Este tipo de sistemas Workflow es el segmento más grande en el mercado. En general automatizan procesos de negocios que tienden a ser repetitivos, bien estructurados y con gran manejo de datos.

- **Colaboración.** Los sistemas Workflow que resuelven procesos de negocios donde participa gente para lograr una meta común, son llamadas Workflow de Colaboración. Los Workflow de colaboración estructuran o semi-estructuran procesos de negocios donde participan personas, con el objetivo de lograr una meta común.
- **Administración.** El Workflow Administrativo es aquel que involucra procesos de administración en una empresa tales como órdenes de compra, reportes de ventas, etc. Estos Workflow's se emplean cuando existe una gran cantidad de procesos administrativos dentro de la empresa y es necesaria la distribución de soluciones a diferentes usuarios.

3.6 Gestión de Procesos Software

Dado el éxito que ha tenido los flujos de trabajo para la automatización y la reingeniería de procesos de negocios, ha atraído la atención de la ingeniería de software, donde se realizan esfuerzos para encontrar soluciones flexibles para la gestión de los procesos de desarrollo en sistemas grandes y complejos (Aversano, 2001).

Los procesos de desarrollo de software son un tipo especializado de procesos de negocios en el cual, documentos, información y tareas son pasados de un participante a otro conforme a un conjunto de reglas conocidas como métodos de desarrollo (Barnes y Gray, 2000).

Los procesos software son complejos, y como todo proceso intelectual, se basa en el juicio humano. Debido a la necesidad de juzgar y crear, los intentos para automatizar estos procesos han tenido éxito limitado, esto debido a la gran diversidad de procesos software. No existe un proceso ideal y diferentes organizaciones ha desarrollado enfoques distintos para desarrollar software,

explotando las capacidades del personal así como las características de las aplicaciones que se están desarrollando. Por lo tanto, aún dentro de una misma compañía pueden existir muchos procesos diferentes para desarrollar software (Sommerville, 2015).

El empleo de un paradigma de flujos de trabajo para la ejecución y gestión de los procesos de desarrollo de software ofrece una mayor apertura para interoperar con otros sistemas, crear entornos de ejecución distribuidos, facilitar el monitoreo y gestionar los recursos humanos (Chan y Leung, 1997).

De forma que los Sistemas Gestores de Flujos de Trabajo pueden ayudar a las organizaciones de software a analizar sus procesos actuales, organizar sus tareas, controlar y monitorear su progreso, y administrar su calidad técnica (Allen, 2001).

Estos sistemas permiten que una organización de software construya un modelo automatizado del marco de trabajo común del proceso, de los conjuntos de tareas y las actividades. El modelo puede analizarse para determinar el flujo de trabajo típico y examinar las estructuras del proceso alternativas que podrían conducir a la reducción del tiempo o costo del desarrollo (Sommerville, 2015).

Una vez creado un proceso aceptable es posible localizar, monitorear e incluso controlar todas las tareas de ingeniería del software definidas como una parte del modelo del proceso. Cada miembro del equipo de software puede emplear herramientas para la elaboración de una lista de verificación de las tareas de trabajo que se deben desarrollar, los productos de trabajo que son imperativos en producir, y las actividades para el aseguramiento de la calidad que deben realizarse.

Los Sistemas Gestores de Flujos de Trabajo pueden coordinar el uso de otras herramientas de la ingeniería del software asistida por computadora que sean apropiadas para una tarea de trabajo particular. En algunos casos incorporan tareas de gestión del proyecto, como estimación, itinerario, rastreo y control

(Aversano et al., 2001). Son más flexibles, y proporcionan a los gestores de proyectos de software cierto grado de libertad para decidir el nivel de detalle en la especificación de un proceso software (granularidad) para adoptar en la realización de un sistema y seleccionar de entre varios modelos de procesos (cascada, iterativos, espiral, etc.) para realizar distintos componentes del sistema.

Aunque los Sistemas Gestores de Flujos de Trabajo pueden ayudar a algunas actividades del proceso, no se prevé, al menos en los próximos años, de una automatización mayor en el diseño creativo del software realizado por los ingenieros relacionados con el proceso software (Sommerville, 2015).

Chan y Leung (1997) establecen que es necesario mantener algunas consideraciones con los sistemas Workflow actuales:

- Ejecución de procesos software de forma automática,
- Reusabilidad de actividades, estructuras, funcionamientos,
- Abstracción para modelar controles y datos,
- Expresividad para facilitar la modelación,
- Monitoreo del desempeño,
- Flexibilidad para realizar cambios de forma dinámica,
- Consistencia en los datos compartidos, manejo de transacciones,
- Verificación para detectar errores en la definición de procesos,
- Control de versiones,
- Integración de herramientas,
- Distribución de procesos entre el personal,
- Datos de estado para la gestión.

3.7 Esquema de Implementación

La implementación de un Sistema Gestor de Flujos de Trabajo en una empresa es similar a otros proyectos de tecnologías de información. Se debe establecer

el ámbito, realizar un plan de implementación, desarrollar la tecnología, implementarla y evaluarla antes de ponerla en uso. Los principales pasos son (Allen, 2001):

1. **Definir el ámbito del proyecto.** Identificar los límites, iniciar en un departamento de misión no crítica. Identificar el tipo de sistema Workflow a ser implementado. Establecer recursos
2. **Planeación de la implementación.** Se presta atención a los detalles, se pretende entender claramente cómo los procesos de negocios trabajan, generalmente mediante entrevistas. Se identifica qué procesos pueden ser mejorados mediante el uso de los sistemas Workflow.
3. **Desarrollo de la aplicación.** Puede considerarse una parte complicada, pero la tecnología de hoy en día para Sistemas Workflow ha tenido numerosos avances, lo que ha permitido generar herramientas sencillas para crear definiciones de procesos y ejecutarlas en entornos. Además que existen API's que ayudan al desarrollador acceder a datos externos.
4. **Evaluación.** Esta actividad no es diferente a las de otros proyectos de implementación de software. Se define un plan de pruebas que cubre todas las eventualidades posibles. Es importante que se tenga un control en todo momento de la gestión de las reglas.
5. **Pre-implementación.** Es importante establecer un plan de capacitación para preparar al personal en el uso del sistema Workflow. Los cambios se dan no solo en la forma de realizar los procesos sino también en la concepción de los mismos.
6. **Post-implementación.** Una vez que la tecnología ha sido liberada es necesario monitorear la eficiencia de los nuevos métodos y aprender de la experiencia para que esta sea incrementada. La automatización crea eficiencia pero es necesario rediseñar los procesos para obtener los máximos beneficios de la combinación de estos elementos.

4. Lecciones Aprendidas

En los últimos años y a nivel mundial, los sistemas Workflow han generado gran interés por dos razones principales: una económica y otra tecnológica. La económica se basa en el reconocimiento que debido a la globalización de los mercados, el ambiente competitivo de negocios actual necesita la automatización de todas las actividades y procesos de trabajo, y no solo la automatización de algunas tareas individuales. La razón tecnológica resulta de

la aparición de nuevos ambientes computacionales, capaces de integrar varias aplicaciones que antes operaban de manera independiente.

A pesar de que organizaciones de software utilizan procesos software diferentes para producir el mismo tipo de producto, es posible automatizar estos procesos. La automatización permite:

- Reducción drástica de tiempos de ciclo, por la agilización de las tareas administrativas y minimización de errores ,
- Mayor integración de las actividades del proceso ,
- Facilita la comunicación y coordinación entre las personas que realizan las tareas,
- Mejora el flujo de información asociado a las actividades del proceso,
- Facilita su control y medición de resultados.

Es decir la automatización mejora el rendimiento, reduce costos y plazos, minimiza errores y garantiza mayor calidad en el software desarrollado.

Los Sistemas Gestores de Flujos de Trabajo automatizan diferentes aspectos del flujo de la información: direccionan los trabajos en la secuencia correcta, proveen acceso a datos y documentos, y gestionan ciertos aspectos de la ejecución de un proceso. Mejoran la eficiencia y la estandarización de los procesos, contribuyendo a la reducción de costos dentro de una empresa, pero más importante aún es que es que permiten obtener un mayor conocimiento de los procesos, y así establecer estrategias para mejorar la calidad.

La gestión de los procesos software con estos sistemas puede monitorear el estado actual de las actividades y observar cómo evolucionan. Así, es posible detectar y solucionar situaciones que requieren tiempos no planificados (decisiones críticas). También permiten asignar tareas de forma automática mediante la definición de roles, así como establecer recursos de información que serán necesarios para poder realizarlas.

Los sistemas Workflow incorporan elementos para control de secuencias de tareas, quiénes realizan dicha secuencia, así mecanismos de control y monitoreo.

Los sistemas Workflow utilizan generalmente una de dos arquitecturas:

1. Formas electrónicas que proporciona una interfaz para tareas. Estas formas son transportadas por todas las fases de las tareas usando mensajes. Este tipo de productos permiten crear aplicaciones que son utilizadas para realizar un trabajo.
2. Aplicaciones externas que proporcionan interfaz para tareas. El entorno de proceso es controlado por un motor Workflow, el cual conserva un registro del progreso de cada instancia del proceso. Este tipo de productos proporcionan una gestión sofisticada del proceso.

5. Conclusión

Los Sistemas Gestores de Flujos de trabajo permiten el control de los procesos software y al mismo tiempo disminuyen la necesidad de intervenir para su gestión. Esto permite que haya menos retrasos en los tiempos destinados a la entrega de un producto o pérdidas como consecuencia de esos retrasos. Al mismo tiempo incrementa la calidad del servicio al responder de una forma más rápida y con el mejor personal disponible.

Además disminuye los costos de entrenamiento debido a que el trabajo puede ser guiado a través de complejos procedimientos, permite una disminución de los costos de gestión, incrementando el control y permitiendo a los administradores concentrarse en el apoyo a los empleados y manejar casos especiales en vez de rutinarios.

Los sistemas Workflow requieren ciertas condiciones para que puedan ser implantados correctamente:

1. Los procesos deben tener tareas explícitas,
2. Reglas para determinar la lógica de transiciones entre tareas,
3. Las tareas utilizan recursos digitales de información,
4. Las tareas deben ser comunicadas a los trabajadores,

5. Hay la necesidad de un proceso de control.

Los elementos básicos de un Sistema de Workflow son

1. Herramientas de análisis, modelado y definición de procesos de negocio. Se representa la secuencia de actividades y de información de cada flujo (Diagramas de flujo de datos).a. Descripción de procesos mediante un lenguaje formal de definición. Incluye. Actividades elementales, operaciones asociadas, reglas de cada etapa y como es la transición entre etapas.
2. Motor del Sistema de Workflow: Interpreta las definiciones de los procesos y crea un entorno para su ejecución.a. Controla las etapas de ejecución de un procesob. Invoca a la persona o aplicación cada etapac. Distribuye las tareasd. Distintos medios de comunicación: Correo electrónico, mensajería, intercambio de datos, etc.
3. Interfaz del sistema de Workflow: con usuarios y aplicaciones. Para rellenar campos de un formulario o manipular un documento.a. Controla la interacción del motor con el usuario o aplicación transfiriendo los datos necesarios en cada casob. Gestiona las colas de trabajo (tareas pendientes, ...)

El modelo de referencia WfMC tiene varios propósitos para el desarrollo de nuevos sistemas Workflow:

1. Ofrece un vocabulario común para describir los procesos de negocios y varios aspectos de tecnologías de soporte que facilitan la automatización. Esto proporciona una base esencial para discusiones sobre cómo puede construirse un sistema Workflow.
2. Una descripción funcional de los componentes de software claves para un Sistema de Gestión de Flujos de Trabajo y cómo deben interactuar. Todo desarrollado con una tecnología neutra de manera que el modelo sea independiente de cualquier arquitectura particular de producto así como tecnología de implementación.
3. La definición en términos funcionales (o abstractos), de la interfaz entre varios componentes de software claves debe facilitar el intercambio de información en una forma estandarizadas, esto es permitiendo el intercambio de información entre productos, cinco interfaces ha sido definidas.

Un sistema Workflow da a una empresa la posibilidad de automatizar sus procesos, reducir costos, y mejorar servicios. Organizaciones que no hayan evaluado esta tecnología podrían encontrarse en desventaja en un futuro.

Referencias

Allen, R. (2001). Workflow: An Introduction. Workflow Handbook 2001. Recuperado de http://www.wfmc.org/information/Workflow-An_Introduction.pdf

Aversano L., de Lucia A., Stefanucci, S. y Betti S. (2001). Introducing Workflow Management in Software Maintenance Processes. 17th IEEE International Conference on Software Maintenance (ICSM'01). Recuperado de <http://doi.ieeecomputersociety.org/10.1109/ICSM.2001.972757>

Barnes, A. y Gray, J. (2000). COTS, Workflow, and Software Process Management: An Exploration of Software Engineering Tool Development. Australian Software Engineering Conference. Recuperado de <http://doi.ieeecomputersociety.org/10.1109/ASWEC.2000.844579>

Berander P. y Wohlin, C. (2003). Identification of Key Factors in Software Process Management - A Case Study. International Symposium on Empirical Software Engineering (ISESE'03). Recuperado de <http://doi.ieeecomputersociety.org/10.1109/ISESE.2003.1237992>

Berger, M., Ellmer, E., Quirchmayr, G. y Zeitlinger A. (1997). Evaluating Workflow Management Systems. 8th International Workshop on Database and Expert Systems Applications (DEXA '97). Recuperado de <http://doi.ieeecomputersociety.org/10.1109/DEXA.1997.617325>

Casati, F., Ceri, S., Pernici, B., y Pozzi, G. (1996). Workflow Evolution. Proceedings of the 15th international Conference on Conceptual Modeling (Octubre 07 - 10, 1996). B. Thalheim, Ed. Lecture Notes In Computer Science, vol. 1157. Springer-Verlag, London, 438-455.

Chan, D. y Leung K. (1997). A Workflow Vista of the Software Process. 8th International Workshop on Database and Expert Systems Applications (DEXA '97). Recuperado de <http://doi.ieeecomputersociety.org/10.1109/DEXA.1997.617237>

Halliday, J., Shrivastava, K. y Wheeler, S. (2001). Flexible Workflow Management in the OPENflow System. Fifth IEEE International Enterprise Distributed Object Computing Conference. Recuperado de <http://doi.ieeecomputersociety.org/10.1109/EDOC.2001.950425>

Hollingsworth, D. WfMC. (1995). Workflow Reference Model, white paper. The Workflow Reference Model (WfMC-TC-1003). Recuperado de <http://www.wfmc.org/standards/docs/tc003v11.pdf>

Microsoft (2016). Introducción a los flujos de trabajo. Recuperado de <https://support.office.com/es-es/article/Introducci%C3%B3n-a-los-flujos-de-trabajo-15d0d459-cce8-4d8b-b34f-9ab6cda1075b#bm2>

OMG (2011). Business Process Modeling Notation Specification. Recuperado de <http://www.omg.org/spec/BPMN/2.0/PDF>

Plesums C. (2002). An Introduction to Workflow. Recuperado de http://www.wfmc.org/information/introduction_to_workflow02.pdf

Pressman, R. (2014). Software Engineering: A Practitioner's Approach. McGraw Hill, 8a. edición, EUA.

Prior, C. (2003). Workflow and Process Management. Maestro BPE Pty Limited, Australia. Recuperado de http://www.wfmc.org/information/Workflow_and_Process_Management.pdf

Sommerville I. (2015). Software Engineering. Pearson, 10a Edición, EUA.

Notas biográficas:



Víctor Hugo Menéndez Domínguez Víctor Hugo Menéndez Domínguez es Doctor en Tecnologías Informáticas Avanzadas por la Universidad de Castilla-La Mancha, España. Es Profesor Titular en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán, México. Su trabajo de investigación se centra en temas relacionados con repositorios de documentos digitales, la representación del conocimiento y la gestión de Objetos de Aprendizaje.



María Enriqueta Castellanos Bolaños María Enriqueta Castellanos Bolaños es Maestra en Gestión de Tecnología de Información por la Universidad Anáhuac Mayab, México. Es Profesora de tiempo completo en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán, México. Su trabajo de investigación se centra en temas relacionados con la gestión del conocimiento y la Ingeniería Web.



Esta obra está bajo una licencia de Creative Commons
Reconocimiento-NoComercial-CompartirIgual 2.5 México.