

Circuitos de documentos configurables. Aproximación a un modelo de datos y algoritmo de implementación.

Mgter. Marciszack, Marcelo
Ing. Quinteros, Sergio
Ing. Muñoz, Roberto Miguel
Ing. Medina, Oscar Carlos

Universidad Tecnológica Nacional, Facultad Regional Córdoba

Abstract

El propósito del trabajo es corroborar la viabilidad de gestionar circuitos de documentos configurables. Para lo cual se realizó el diseño funcional de un módulo de software basado en el supuesto de que los procesos de un sistema de gestión documental pueden representarse con un conjunto simplificado de objetos de flujo del estándar Notación de Modelado de Procesos (BPMN). A los fines de que un usuario final de este módulo, pueda parametrizar los distintos circuitos de documentos de su organización desde una interfaz web amigable, se definen tipos de acciones para determinar anticipadamente el estado y el próximo destino posible del documento en todas sus etapas de vida: creación, movimientos y cierre.

Palabras Clave

BPM, BPMN, BPMS, Gestión por Procesos, “workflow”, flujo de proceso, configuración.

Introducción

En la actualidad el modelado de procesos de negocios tiene unificado y consensuado sus lineamientos en una metodología que se llama Gestión por Procesos, más conocida por su sigla en inglés BPM.

Se dice que un proceso de negocio responde a un flujo de trabajo (“workflow”) que es un conjunto de acciones ordenadas en una secuencia determinada, en el cual las tareas se ejecutan en forma manual o automatizada.

Partiendo de la premisa que los procesos de la gestión documental de una organización también responden a flujos de trabajo predefinidos, para poder implementarlos mediante un módulo de software, se modelan bajo el nombre de circuitos de

documentos que representan un conjunto de pasos ordenados con dependencia entre sí, que dan origen a un documento e indican el trayecto que debe seguir el mismo en cada una de sus “edades”, incluyendo su archivado de gestión, guarda histórica o destrucción.

Para el modelado de los circuitos de documentos se optó por una BPMN simplificada. BPMN significa Notación de Modelado de Procesos de Negocios y es una técnica que proporciona en forma estándar la representación de “procesos de negocio, tanto para fines descriptivos de alto nivel, como para detallados y rigurosos entornos de software orientados a procesos” [1].

La BPMN que se utiliza en los circuitos de documentos mencionados permite indicarle al sistema qué acción sugerir en cada paso del proceso mediante objetos de flujo que son los elementos principales que definen la estructura y el comportamiento del proceso. Para esta propuesta, los objetos de flujo se denominan Tipo de Acción y permiten asignarles nombres más cercanos del lenguaje coloquial a tareas habituales del negocio como lo son la apertura, el pase y la numeración de documentos.

Con estos elementos, el esquema conceptual de un módulo de software debería dar respuestas principalmente a dos preguntas:

¿Se puede adaptar la nomenclatura BPMN para representar procesos de gestión documental de una manera más simple?.

Si dicha respuesta fuera positiva, con la mencionada nomenclatura BPMN simplificada, ¿es factible que un usuario final defina procesos de documentos configurando los pasos y tipo de acciones de cada circuito?.

Elementos del Trabajo y metodología

A continuación se exponen los principales componentes del modelo planteado.

a) Pasos de un circuito de documento:

Un circuito de documentos está compuesto por pasos, cada uno de ellos asociados a un tipo de acción y debe incluir al menos un paso con el tipo de acción “Apertura”. No hay límite de pasos por circuito.

A modo de ejemplo, se detallan dos circuitos de documentos:

I- Circuito de numeración individual: la numeración de los documentos la realiza el área iniciadora del proceso y luego se realizan los pases entre áreas de la organización.

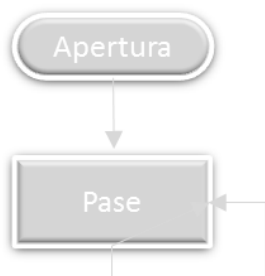


Figura 1: Diagrama de flujo del Circuito de numeración individual

Paso 01: Apertura	
Tipo de acción:	Apertura
Áreas origen:	Todas
Áreas destino:	Todas
Pasos predecesores:	00
Pasos sucesores:	02
Ocurrencia múltiple:	No
Obligatorio:	Si
Paso 02: Pase	
Tipo de acción:	Pase
Áreas origen:	Todas
Áreas destino:	Todas
Pasos predecesores:	01, 02
Pasos sucesores:	02
Ocurrencia múltiple:	Si
Obligatorio:	No

Tabla 1: Circuito de numeración individual

II- Circuito de numeración centralizada: los documentos se numeran en forma centralizada en una sola área de la organización, y luego pueden hacerse los movimientos (pases) a otras áreas.

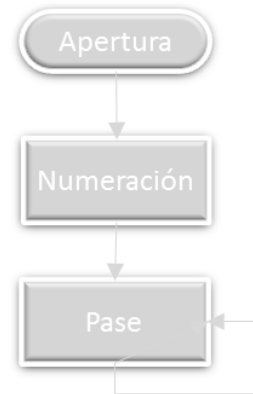


Figura 2: Diagrama de flujo del Circuito de numeración centralizada

Paso 01: Apertura	
Tipo de acción:	Apertura
Áreas origen:	Todas
Áreas destino:	Todas
Pasos predecesores:	00
Pasos sucesores:	02
Ocurrencia múltiple:	No
Obligatorio:	Si
Paso 02: Numeración	
Tipo de acción:	Numeración
Áreas origen:	Todas
Áreas destino:	Mesa de Entradas
Pasos predecesores:	01
Pasos sucesores:	03
Ocurrencia múltiple:	No
Obligatorio:	Si
Paso 03: Pase	
Tipo de acción:	Pase
Áreas origen:	Todas
Áreas destino:	Todas
Pasos predecesores:	02, 03
Pasos sucesores:	03
Ocurrencia múltiple:	Si
Obligatorio:	No

Tabla 2: Circuito de numeración centralizada

b) Modelo de datos propuesto:

A continuación se muestra a modo de ejemplo una implementación del presente modelo y se enuncian los datos que conforman un circuito de documento.

MUNICIPALIDAD >> SIRAD
 CIUDAD DE CÓRDOBA

Bandeja de documentos - Usuario superadministrador
 Entidad Mesa General de Entr...

Inicio Documentos Persona Configuración

Circuitos

Ver

Circuito (*): General Numeración Centralizada

Descripción (*): Procedimiento admin. municipal

Normativa: Ordenanza 12345/2013

Vigencia (*): 01/01/2013 31/12/9999

Pasos del circuito

Nro.	Descripción	Tipo Acción	Entidad Origen	Entidad Destino	Predecesores	Sucesores	Múltiple	Obligatorio
01	Apertura	APERTURA	Todas	Todas	00	02	NO	SI
02	Numeración	NUMERACIÓN	Todas	Mesa General de Entradas, Aforos y Archivo General	01	03	NO	SI
03	Pase	PASE	Todas	Todas	02, 03	03	SI	NO

Volver

2013 © Municipalidad de Córdoba | Sistema inteligente rector de la administración documental municipal

Figura 3: Pantalla de configuración de circuito de documentos implementado en la aplicación SIRAD

- **Número de Paso:** Identificador del paso dentro del circuito. No indica el orden de ejecución de este paso sino que el mismo está dado por los pasos predecesores y sucesores. Este dato se asigna y mantiene actualizado automáticamente en forma secuencial correlativa ascendente.
- **Descripción de Paso:** Texto que relata brevemente la función que se lleva a cabo en el paso.
- **Tipo de acción:** Tipo de acción del paso. Representa la tarea que puede realizar el Área Origen en el paso actual de un documento. Se profundiza este punto más adelante al tratar el *Algoritmo de implementación*.
- **Áreas Origen:** Indica las Áreas de la organización que pueden realizar la acción con el documento en el paso actual de un documento. El valor 'Todas' representa a cualquier Área del organigrama vigente.
- **Áreas Destino:** Indica las Áreas de la organización que van a tener asignado el documento en el próximo paso. El valor 'Todas' representa a cualquier Área del organigrama vigente.
- **Pasos predecesores:** Representa a los pasos que le deben anteceder al paso actual de un documento en el circuito. Si tiene el valor '00' indica que es el primer paso.
- **Pasos sucesores:** Representa a los pasos que pueden sucederle al paso actual de un documento en el circuito. Si tiene el valor '00' indica que es el último paso.
- **Ocurrencia múltiple:** Indica si el paso se puede realizar más de una vez en el circuito. Puede asumir el valor 'Si' o 'No'.
- **Obligatorio:** Indica si el paso es de ocurrencia obligatoria u opcional en el circuito. Puede asumir el valor 'Si' o 'No'.

- **Formulario de usuario:** Nombre o “url” de una página web desarrollada específicamente para una funcionalidad que se ejecuta asociada a un paso de tipo de acción *PaseFormulario*. El formulario de usuario no puede recibir parámetros de entrada y debe retornar un único valor de salida lógico (“Verdadero” o “Falso”) que permitirá saber si el paso fue completado o no con éxito.
- **Grupo de campos adicionales:** Página web dinámica, compuesta por un grupo de campos adicionales al conjunto de datos básicos de un documento, que se ejecuta asociada a un paso de tipo de acción *PaseAdicional*. Este dato es equivalente al Formulario de usuario aunque no requiere que se programe pero sí debe configurarse para su ingreso. Esta funcionalidad igual que *PaseFormulario*, no puede recibir parámetros de entrada y debe retornar un único valor de salida lógico (“Verdadero” o “Falso”) que permitirá saber si el paso fue completado o no con éxito.
- **Condición (de *PaseSi*):** Condicional asociada a un paso del tipo de acción *PaseSi* con operadores unitarios, binarios, lógicos, atributos del documento y tablas relacionadas, cuyo valor determinará la ejecución del paso si es “Verdadero”.
- **Lista de pasos:** Lista de pasos anteriores asociados a un paso del tipo de acción *PaseSino* que determina su ejecución si la condición de todos los pasos seleccionados (que deben ser del tipo de acción *PaseSi*) devuelven un valor “Falso”.
- **Días (de Espera):** Condicional asociada a un paso del tipo de acción *Espera* que permite su ejecución si el valor numérico que representa la cantidad de días del plazo se cumple desde la fecha

y hora que fue recibido el documento. También se pueden representar horas usando fracciones de día, siendo 1 hora equivalente a 1/24 día.

c) Paso actual de un documento:

Para poder determinar en qué paso de un circuito se encuentra un documento, se registra el número de paso del circuito en cada movimiento del documento.

De esta forma, cuando el sistema necesite conocer el paso actual de un documento, debe buscar el último movimiento registrado. Si no tiene ninguno, el documento está en el paso sucesor del primero, o sea del tipo de acción *Apertura*. Por ejemplo, si el documento tiene asociado un circuito de numeración individual, se encuentra en el paso 02, ya que es el sucesor del paso que tiene tipo de acción *Apertura*.

Caso contrario, si se encuentra el último movimiento del documento, se está en el paso sucesor del paso registrado en el movimiento, indicando el área destino. Por ejemplo, si el documento tiene asociado un circuito de numeración centralizada y el último movimiento tiene registrado un paso 03, se encuentra nuevamente en el paso 03, ya que es su único sucesor.

Si hubiera más de un paso sucesor en las situaciones anteriormente mencionadas, la identificación del paso la resuelve el actor al indicar el área destino cuando registre el próximo movimiento. Por ejemplo, si el documento tiene asociado una variante del circuito de numeración individual que se describió anteriormente, con dos tipos posibles de pases: el paso 02 ‘Pase Legajo Personal’ con área destino “RR.HH.” y el paso 03 ‘Pase Factura Proveedor’ con área destino “Tesorería”, ambos posteriores al paso 01 ‘Apertura’.

Paso 01: Apertura	
Tipo de acción:	Apertura
Áreas origen:	Todas
Áreas destino:	Todas
Pasos predecesores:	00
Pasos sucesores:	02,03
Ocurrencia múltiple:	No
Obligatorio:	Si
Paso 02: Pase Legajo Personal	
Tipo de acción:	Pase
Áreas origen:	Todas
Áreas destino:	RR.HH.
Pasos predecesores:	01
Pasos sucesores:	04
Ocurrencia múltiple:	No
Obligatorio:	No
Paso 03: Pase Factura Proveedor	
Tipo de acción:	Pase
Áreas origen:	Todas
Áreas destino:	Tesorería
Pasos predecesores:	01
Pasos sucesores:	04
Ocurrencia múltiple:	No
Obligatorio:	No
Paso 04: Pase	
Tipo de acción:	Pase
Áreas origen:	Todas
Áreas destino:	Todas
Pasos predecesores:	02, 03, 04
Pasos sucesores:	04
Ocurrencia múltiple:	Si
Obligatorio:	No

Tabla 3 Variante del circuito de numeración individual

Si el último movimiento del documento fue el paso 01, como se encuentran dos posibles sucesores, el paso 02 y el paso 03, el sistema identificará el paso del documento al momento que se registre el próximo movimiento de acuerdo al área destino que se lo envíe, será 02 si el área destino es “RR.HH.” o será 03 si se lo envía a “Tesorería”. Vale aclarar para este caso, que se puede utilizar una condición que determine automáticamente en el paso 01 la próxima área destino del documento. Se grafica el ejemplo con el siguiente diagrama:

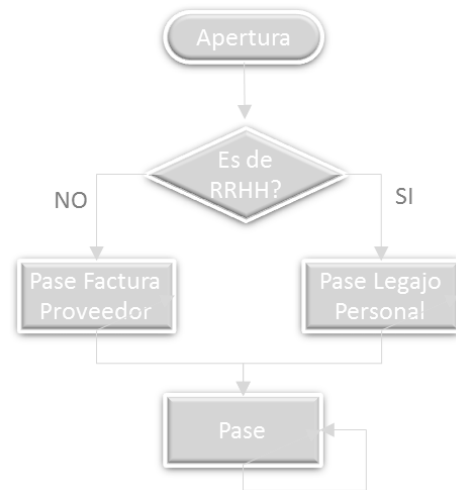


Figura 4: Diagrama de flujo de variante del circuito de numeración individual

d) Algoritmo de implementación:

Finalmente, se presentan las piezas claves de los circuitos de documentos configurables, para lo cual se seleccionó y adaptó un conjunto reducido de objetos de flujo del estándar BPMN en su versión 2.0 [2] y se los llamó tipo de acción, con miras a facilitar la tarea del usuario que tiene la responsabilidad de definir con este módulo los circuitos sugeridos para la gestión de los documentos. Se describe a continuación los mencionados tipos de acción.

- **Apertura:** Creación de documento. El Área Origen indica quien lo puede crear y el Área Destino quién lo va a tener en su bandeja de documentos para poder completar los datos. Este tipo de acción no tiene ningún paso predecesor.
- **Pase:** Movimiento de documento desde el Área Origen al Área Destino. Es una adaptación del evento intermedio de vínculo de la BPMN.
- **Numeración:** Caso particular de movimiento de documento donde el Área Origen es quien hace la apertura del documento y se envía de forma automática y obligatoria al Área Destino para que lo numere. Luego, cuando el Área Destino numera el documento, el sistema lo devuelve automáticamente al Área Origen para

que realice el paso que corresponda de acuerdo al circuito definido.

- PaseFormulario** <formulario de usuario>: Movimiento de documento donde el Área Origen debe completar un formulario diseñado a la medida de las necesidades del área para ese paso del circuito. Si el formulario de usuario se registra correctamente, devuelve un valor “Verdadero” y el sistema envía el documento al Área Destino. El dato <formulario de usuario> es requerido para este tipo de acción e indica el nombre o la “url” del programa. Un nuevo formulario de usuario implica construir nuevas funcionalidades que deben ser programadas por un desarrollador. La extensibilidad que brinda este tipo de acción es la de integrar a circuitos existentes, futuros requerimientos con necesidad de ingreso de nuevos datos por pantalla.
- PaseAdicional** <grupo de campos adicionales>: Movimiento de documento donde el Área Origen debe completar un formulario dinámico conformado por un grupo de campos adicionales para este paso del circuito. Si el formulario dinámico se registra correctamente, devuelve un valor “Verdadero” y el sistema envía el documento al Área Destino. El dato <grupo de campos adicionales> es requerido para este tipo de acción, y representa un conjunto de campos adicionales que se agregan a los datos básicos de los documentos. Un grupo de campos adicionales puede ser configurado por los usuarios finales del sistema a diferencia de un formulario de usuario que requiere que sea programado y publicado en una nueva versión del aplicativo.
- PaseTareaLote** <tarea de procesamiento en lote>: Movimiento de documento donde en el Área Origen se debe ejecutar una tarea de procesamiento en

lote, en ese paso del circuito. Si la tarea de procesamiento en lote se ejecuta correctamente, devuelve un valor “Verdadero” y el sistema envía el documento al Área Destino. El dato <formulario de usuario> es requerido para este tipo de acción e indica el nombre del programa de procesamiento en lote. La diferencia entre *PaseFormulario* y este tipo de acción es que el programa se corre sin requerir interacción alguna con los usuarios.

- PaseSi** <condición>: Movimiento de documento desde el Área Origen al Área Destino siempre y cuando se cumpla una condición. El dato <condición> es requerido para este tipo de acción y posee una sintaxis con operadores y atributos del documento, cuyo valor resultante determinará la ejecución del paso si es “Verdadero”. Se trabajan con tres tipos de operadores al definir una condición: unitarios, binarios y lógicos.

Operadores unitarios: condiciones sobre el propio valor del atributo.	Si es nulo o no	IS (NOT) NULL
	Si está vacío o no	IS (NOT) EMPTY
	Texto coincidente	LIKE
Operadores binarios: comparan el valor de un atributo con respecto a otro.	Igual	=
	Distinto	<>
	Mayor o igual	>=
	Menor	<
Operadores lógicos: comparan dos valores lógicos: “Verdadero” o “Falso”.	Y	AND
	O	OR

Tabla 4: Operadores de una condición

Dos o más acciones del tipo *PaseSi* <condición> permiten modelar lo que en BMPN se denomina elemento de divergencia ya sea como compuerta paralela, porque indica un punto del proceso en la cual los pasos siguientes pueden ser llevadas a cabo en forma concurrente (paralela), o como compuerta inclusiva, porque indica un punto del circuito donde uno o varios caminos son activados.

Se puede ejemplificar el comportamiento de este tipo de acción haciendo una comparación con dos sentencias de programación de uso generalizado.

Modelado de sentencia IF THEN:

- PaseSi <condición=“Verdadero”>
- PaseSi <condición=“Falso”>

Modelado de sentencia DO CASE:

- PaseSi <condición 1= “Verdadero”>
- PaseSi <condición 2= “Verdadero”>
- PaseSi <condición n= “Verdadero”>

- PaseSino <lista de pasos>: Movimiento de documento desde el Área Origen al Área Destino siempre y cuando no se cumpla ninguna condición, o sea que TODAS las condiciones de los pasos enumerados en la lista devuelvan resultado “Falso”. Se agrega un dato al paso del circuito que es requerido para este tipo de acción y permite detallar los números de pasos seleccionados.

Con este tipo de acción se completa la implementación de los elementos de divergencia simplificando la condicional que se requeriría al enunciar el negativo lógico de las condiciones de un conjunto de pasos *PaseSi*.

Para ejemplificar el comportamiento de este tipo de acción, se hace nuevamente una comparación con otras dos sentencias de programación de uso generalizado.

Modelado de sentencia IF THEN / ELSE:

- paso 2: PaseSi <condición es “Verdadero”>
- paso 3: PaseSino <paso 2>

Modelado de sentencia DO CASE/OTHERWISE:

- paso 2: PaseSi <condición 1 = “Verdadero”>
- paso 3: PaseSi <condición 2 = “Verdadero”>
- paso n: PaseSi <condición n = “Verdadero”>

- paso n+1: PaseSino <pasos 2,3,n>

- PaseUnión <lista de pasos>: Movimiento de documento desde el Área Origen al Área Destino siempre y cuando se cumpla que se hayan realizado al menos una vez TODOS los pasos especificados en la <lista de pasos>, dato requerido para este tipo de acción. Los números de paso de la lista pueden ser del 2 a n-1, siendo n el número de paso actual y los pasos a seleccionar deben ser del tipo de acción *PaseSi*.

Este elemento de flujo es equivalente a lo que en BPMN se denomina compuerta paralela convergente permitiendo modelar un patrón de sincronización. La sincronización es un paso en el circuito donde dos o más caminos del proceso se unen en uno solo. Se llama sincronización porque se espera a que todos los caminos entrantes, paralelos o no, se completen antes de continuar con el siguiente paso.

- PaseCircuito <circuito de documento>: Movimiento de documento desde el Área Origen al Área Destino siempre y cuando se cumpla que se hayan realizado al menos una vez TODOS los pasos del circuito especificado en <circuito de documento>, dato requerido para este tipo de acción. Este tipo de acción le brinda modularidad al algoritmo de implementación, ya que se pueden dividir circuitos complejos en sub-circuitos para un tratamiento más adecuado.

- Espera <n días>: Movimiento de documento desde el Área Origen al Área Destino siempre y cuando se cumpla un plazo de n días especificado en el dato <n días>, requerido para este tipo de acción. El valor n es un número decimal, ya que puede expresar plazos en horas, siendo 1 hora equivalente a 1/24 día. El plazo de espera se cumple desde la fecha y hora que fue recibido el documento y, vale aclarar que si esto

ocurre, no se realiza el movimiento automáticamente sino que a partir de ese momento va a agregarse el Área Destino de este paso, si correspondiera, como sugerida para realizar el próximo movimiento.

- Cierre: Movimiento de documento desde el Área Origen a ninguna Área Destino. A partir de este paso no se pueden realizar más movimientos de documento. No se debe confundir con el estado de “Archivado” del documento, este tipo de acción es equivalente a un patrón de terminación de BPMN.

Los patrones de terminación son utilizados para establecer cuándo poner fin a las instancias de un proceso de acuerdo a las condiciones del negocio, en este caso en particular, el cierre finaliza el circuito de un documento. Por ejemplo son acciones de cierre el archivado histórico o la destrucción del documento luego de su valoración de guarda.

Como puede apreciarse, el diseño de la solución consiste en definir una secuencia finita, aunque recursiva, de pasos para un circuito de documento, asociando a cada uno de ellos principalmente un tipo de acción, área origen y área destino, lo que permite determinar un flujo de proceso cuando se realicen los movimientos de documentos en la organización.

Resultados

El diseño puesto en consideración fue construido e integrado al módulo Configuración de Documentos del proyecto SIRAD (Sistema Rector de la Administración Documental municipal) desarrollado por el CIDS (Centro de Investigación y Desarrollo de Sistemas) de la U.T.N. F.R.C. para la Municipalidad de Córdoba.

El objetivo de esta solución informática es “apoyar la operatoria de Mesa General de Entradas, Aforos y Archivo General, en lo referente al seguimiento de expedientes,

notas y cualquier documento que sea tramitado en la Municipalidad de Córdoba, como así también dar soporte a la identificación numérica de documentación de la Administración Pública Municipal y la registración de sus movimientos, desde su iniciación hasta su archivo o destrucción” [3].

El sistema SIRAD está organizado en los siguientes módulos:

- Administración de entidades
- Administración de personas
- Configuración de documentos
- Seguimiento de documentos
- Informes y estadísticas
- Vista del ciudadano
- Parámetros generales

Las relaciones entre los distintos módulos se muestran en el siguiente esquema conceptual [3]:



Figura 5: Módulos de SIRAD

A la fecha de redacción del artículo (Julio de 2013), SIRAD se encuentra en fase de pruebas “beta” y con una puesta en producción prevista para Octubre del mismo año. Como el procedimiento actual es soportado por un sistema legado de AS/400, se migran más de 1.000.000 documentos y de 4.000.000 de movimientos para que puedan continuar trabajando con ellos alrededor de 3.000 usuarios pertenecientes a 360 áreas de la organización.

Para esta primera salida en vivo de la aplicación se configuraron dos circuitos de documentos, uno de numeración individual para las notas y otro de numeración

centralizado para los expedientes, como los que se ejemplifican en las *Tablas y Figuras 1 y 2* respectivamente. El flujo de proceso de estos dos circuitos fue definido utilizando solamente tres tipos de acciones: Apertura, Numeración y Pase.

Con esta parametrización se satisfacen los requerimientos de la totalidad de procedimientos de los documentos que hoy en día gestiona la organización, pero se “abre una puerta” para la automatización y mejora continua de nuevos procesos siguiendo una estrategia análoga a la de los circuitos generales y sumando gradualmente tipos documentales (decreto, resolución, ordenanza, memorándum, etc.) y el resto de los tipos de acciones detalladas en el algoritmo de implementación. Por ejemplo se prevé para la configuración del circuito del decreto una variante del circuito de numeración individual con áreas acotadas de origen y destino. También se cuenta con los tipos de acción *PaseSi* y *PaseSino* como derivadores automáticos de los circuitos de aprobación permitiendo la formalización de las reglas de negocio, acorde a la estructura jerárquica de la organización. Por último, los tipos de acción *PaseAdicional*, *PaseFormulario*, *PaseTareaLote* y *PaseCicuito* le brindan a la aplicación las cualidades de extensibilidad y dinamismo para adaptarse a nuevos requerimientos y cambios de normativa.

Discusión

Según Pablo Trilles Farrington, Vicepresidente AuraPortal: “Con un poco de perspectiva es difícil concebir que un BPMS no contenga Gestión Documental y de Contenidos Web” [4]. BPMS significa Sistema BPM o sea de Gestión por Procesos. Coincidente con esta opinión es el planteo inicial de este trabajo, ya que no busca ser original en el uso de BPMN para definir circuitos de documentos, sino en el modo “amigable” de presentar las herramientas de BPM a los usuarios finales de una aplicación de gestión documental,

que cuenta con algunas funcionalidades de BPMS.

Asimismo, puede ser cuestionada la decisión de implementar el modelo con un desarrollo propio en vez de integrar un componente de software libre, como lo hizo la Univesitat de València que utilizó el Bonita workflow engine como tecnología de soporte de los flujos de trabajo para el sistema SGIC-UV (Sistema de Garantía Interna de Calidad de centros de titulaciones-Universitat de València).

Cabe preguntar, ¿serían válidos ambos caminos si arriban al mismo destino con costos y tiempos acordes a lo requerido por el cliente? A decir de Alfredo Pérez Boulosa, Amparo Chirivella Ramón y Vicente Cerverón Lledo: “El motor de flujos de trabajo será el elemento que se encarga de gestionar a qué usuario le corresponde realizar cada tarea (según la descripción del proceso) y cuando la haya completado y en función de determinadas condiciones, determinar cuál es la siguiente tarea que debe realizarse (y quién debe hacerlo)” [5]. O sea, es válido si se cumple la función de gestionar los flujos de proceso. Continuando en esta línea de análisis, se pregunta Bernhard Hitpass: “¿Podría el ingeniero de procesos implementar en TI la solución sólo con la información del diagrama? La respuesta es clara, por supuesto que no” [6]. Son necesarias la interfaz de usuario y las entidades de datos, para este caso, los documentos. Tener la orquestación de los circuitos de documentos en una capa separada de la interfaz de usuario otorga una mayor flexibilidad para los cambios. Finalmente, citando a Bernhard Hitpass de nuevo, esta vez junto a Jakob Freund y Bernd Rucke: “En un principio en un proceso hay que hacer ciertas cosas (actividades), pero a lo mejor solo bajo ciertas condiciones (Gateways) y además pueden ocurrir cosas (eventos). A estos objetos se les denomina en BPMN objetos de flujo...” [7], lo que nos da un indicio del amplio espectro de herramientas del BPMN. Por ello, es conveniente resaltar

que falta cotejar la completitud del conjunto de tipos de acción seleccionado con la implantación de SIRAD, y a posteriori, si se desea replicar el modelo, para asegurarse que contemple la representación de todos los circuitos de documentos que el cliente requiera. Por ejemplo, ¿por qué no se incluyó un tipo de acción *CobroDeAforo* si tiene igual relevancia que la *Numeración* de documentos en los procesos de Mesa de Entradas?. El criterio aplicado fue que la numeración es un paso obligatorio a diferencia del aforo que es un requisito dentro del tipo de acción *Numeración* basado en los datos del documento. Parafraseando a Albert Einstein, “Si Usted va a describir la verdad deje la elegancia al sastre”, en este análisis se propone un “vestuario” básico y necesario para describir procesos de gestión documental, con el objetivo de facilitar esta tarea al usuario final de la aplicación.

Conclusión

A la primera pregunta planteada en la Introducción: *¿se puede adaptar la nomenclatura BPMN para representar procesos de gestión documental de una manera más simple?*, la respuesta es afirmativa. Porque el subgrupo de objetos de flujo de BPMN que se seleccionaron, es el que dio lugar a los tipos de acción, ejes centrales del algoritmo de implementación de cada circuito. Como ejemplos de particularización se pueden mencionar al *Pase* que es un evento intermedio de vínculo y *Numeración* que es una tarea frecuente en las aplicaciones de gestión documental.

En cuanto a si *¿es factible que un usuario final defina procesos de documentos configurando los pasos y tipo de acciones de cada circuito con esta nomenclatura simplificada?*, se vislumbra una respuesta en dos etapas: primero la construcción y luego la puesta en marcha. A la fecha, para el desarrollo del módulo que hace las veces de motor de flujo de procesos, los resultados son positivos, ya que permitió su

programación y prueba en forma gradual, y con rendimiento notable de los programadores, en relación al tiempo de desarrollo según complejidad del caso de uso. Esto se debe a que la interfaz de configuración de un circuito es similar a la de cualquier entidad con relación encabezado y detalle, estando compuesto el detalle por el modelo de datos descrito en este artículo. Aún falta la retroalimentación de los usuarios finales luego de su puesta en producción, y su operatoria en fase de mantenimiento cuando requieran crear un nuevo circuito de documento o modificar uno existente. Esta experiencia se va a intentar anticipar con las pruebas de validación y aceptación, evaluando el manejo del sistema y simulando cambios en distintos escenarios ficticios. Se tiene en cuenta además que la aceptación de una nueva solución informática se refuerza con el acompañamiento en el cambio organizacional, la comunicación institucional, las capacitaciones y el soporte técnico a los usuarios finales.

En definitiva, lo expuesto anteriormente no constituye una contribución revolucionaria a la Gestión por Procesos, sin embargo pretende darle un enfoque distinto a la representación de los procesos de gestión documental para que, aprovechando las ventajas de esta metodología, se puedan configurar de manera más sencilla que utilizando la BPMN tradicional.

Agradecimientos

A las respectivas familias, *nuestros BMP*, bienes más preciados y a los integrantes del proyecto SIRAD que conformamos colegas del CIDS de U.T.N. F.R.C. y empleados de la Municipalidad de Córdoba.

Referencias.

- [1] Stephen A. White, Phd - Derek Miers: Guía de Referencia y Modelado BPMN. Comprendiendo y utilizando BPMN. Future Strategies Inc. - Lighthouse Point, Florida – Ed. digital en español, USA; 2009.
- [2] OMG Object Management Group, Inc.: Business Process Model and Notation (BPMN) Version 2.0 - OMG Final Adopted Specification - OMG Document Number: formal/2011-01-03 - January 2011 (versión actual, ed. digital).

[3] Documentación funcional del proyecto SIRAD SIRAD (Sistema Rector de la Administración Documental municipal) desarrollado por el CIDS (Centro de Investigación y Desarrollo de Sistemas) de la U.T.N. F.R.C. para la Municipalidad de Córdoba; 2013.

[4] Pablo Trilles Farrington: La evolución del BPM. El libro del BPM 2011. Centro de Encuentro BPM, s.l (Club BPM). Madrid, España; 2011.

[5] Alfredo Pérez Boullosa – Amparo Chirivella Ramón – Vicente Cerverón Lledo: Seguimiento de las titulaciones y Sistema de Garantía de Calidad. Importancia de la herramienta informática. Unitat de Qualitat – Universitat de València. XII Foro de Almagro. El papel de los indicadores en el seguimiento y en los sistemas de garantía interna de la calidad. Universidad de Castilla-La Mancha, España; 2011.

[6] Bernhard Hitpass: BPM. Business Process Management Fundamentos y Conceptos de

Implementación. BPMCenter – Camunda 2ª Ed. hispana, Chile; Enero 2013.

[7] Jakob Freund – Bernd Rucker – Bernhard Hitpass: BPMN 2.0. Manual de Referencia y Guía. BPMCenter – Ed. hispana, Chile; Marzo 2011.

Datos de Contacto:

Mgter. Marcelo Marciszack,
marciszack@gmail.com - *Ing. Sergio Quinteros,*
ser.quinteros@gmail.com - *Ing. Roberto Miguel*
Muñoz, rmunoz@systemas.frc.utn.edu.ar - *Ing.*
Oscar Carlos Medina, oscarcmolina@gmail.com.
Universidad Tecnológica Nacional -Facultad
Regional Córdoba – Departamento Ingeniería en
Sistemas de Información – Dirección: Maestro
López esq. Av. Cruz Roja Argentina, Ciudad
Universitaria – (5000) Córdoba.