

Un caso de estudio para la adopción de un BPMS

Javier Luis Cánovas Izquierdo¹, Óscar Sánchez Ramón¹,
Jesús García Molina¹, Carlos Castillo Alarcón²

¹Facultad de Informática, Universidad de Murcia
{jlcanovas, osanchez, jmolina}@um.es

²Sinergia Tecnológica
carlos.castillo@sinergiatec.com

Resumen

Los Sistemas de Gestión de Procesos de Negocio (BPMS) proporcionan un nuevo paradigma orientado a procesos para crear aplicaciones para la gestión de las organizaciones. Un BPMS ejecuta modelos de procesos de negocio y proporciona herramientas para la simulación, monitorización y ajuste de los procesos de negocio. Este paradigma se puede enriquecer con el uso del Desarrollo Dirigido por Modelos para resolver las cuestiones de integración de servicios en un proceso de negocio.

En la actualidad, hay una carencia de casos de estudio sobre BPM y una gran confusión sobre qué es y qué no es un BPMS. En este artículo se presenta una solución BPM para un caso de estudio de un servicio de la Administración Pública Regional, así como un análisis de algunas cuestiones sobre BPMS que se plantean con frecuencia. Para la integración de servicios en el proceso de negocio se muestra una aproximación basada en transformaciones de modelos.

Palabras clave: BPM, BPMS, MDD.

1. Introducción

Los Sistemas de Gestión de Procesos de Negocio (BPMS, *Business Process Management System*) son plataformas software que permiten el modelado, despliegue y seguimiento de los procesos de negocio de una organización por parte de desarrolladores, analistas del negocio y administradores del sistema. Desde su aparición, a principios de esta década, el mercado de los BPMS ha experimentado un continuo crecimiento, como señala un informe reciente de Gartner [1],

que estima una tasa de crecimiento compuesta anual del 24% entre 2006 y 2011.

En esta década también ha emergido el paradigma del Desarrollo de Software Dirigido por Modelos (MDD, *Model Driven Development*) como una nueva forma de abordar la creación de software a partir de lenguajes de modelado específicos del dominio. Estos lenguajes permiten aplicar un nivel de abstracción mayor que los lenguajes de programación tradicionales, y la generación automática de código a partir de modelos gráficos o especificaciones textuales expresados con dichos lenguajes. En realidad, el término “Desarrollo Dirigido por Modelos” no se refiere a un único paradigma sino a un conjunto de paradigmas tales como MDA (sin duda, el más conocido), las Factorías de Software, o el Desarrollo Específico del Dominio.

En la actualidad, nuestro grupo de investigación colabora con la empresa Sinergia Tecnológica (perteneciente al grupo IT Deusto) en un proyecto piloto destinado a adquirir conocimientos sobre las herramientas BPMS con el fin de adoptarlas para el desarrollo de aplicaciones. Como caso de estudio se ha elegido la aplicación de un BPMS en un servicio de la Administración Pública Regional. Hasta el momento se ha completado la fase de modelado y despliegue, y falta por completar el período de pruebas y evaluación de resultados.

El objetivo de este trabajo es presentar la solución BPM estudiada para el mencionado caso de estudio y analizar algunas cuestiones que surgen cuando se plantea utilizar BPM, como lo es su relación con MDD. El proceso de negocio modelado interactúa con servicios externos, que han sido generados automáticamente mediante técnicas MDD y que acceden a la lógica de

negocio existente en un sistema heredado (*legacy system*). Por tanto, el caso de estudio también ha permitido analizar las capacidades de integración de BPM.

Este trabajo se ha organizado del siguiente modo. En el siguiente apartado se mostrará una breve descripción del problema abordado. En el tercer apartado se presentan los argumentos que justifican la elección del sistema BPM, se describe el proceso de desarrollo aplicado con este sistema, que es una adaptación de un proceso más general para BPMS, y se comenta cómo se ha integrado la lógica de negocio existente. A continuación se valoran algunos aspectos de los BPMS, fruto de la experiencia obtenida con el caso de estudio. Finalmente se presentan unas conclusiones.

2. Descripción del problema

Una de las principales actividades que realiza el Servicio de Empleo y Formación de la Región de Murcia (SEFCARM) es la concesión de subvenciones a entidades colaboradoras para la realización de cursos de formación. La Figura 1 muestra el ciclo de vida de una subvención, que consta de las siguientes fases: alta de la entidad y del curso a impartir, creación de solicitud de la subvención, baremación de la solicitud, inicio del curso y realización del mismo, seguimiento y liquidación.

En la actualidad, la empresa Sinergia Tecnológica está abordando la creación de un nuevo sistema que gestione el ciclo de vida anteriormente citado. Ha comenzado con el desarrollo de las aplicaciones para las fases de seguimiento y liquidación ya que existía un sistema anterior que implementaba la primera fase del ciclo, la cual comprende desde el registro de la entidad y el curso, hasta la concesión de la subvención. Este sistema heredado implementaba la lógica de negocio en PL/SQL. Como el nuevo sistema se desarrollaba sobre la plataforma Java, era necesario crear unos *wrappers* para acceder al código PL/SQL. En [2] se describe cómo se generaron estos *wrapper* aplicando técnicas MDD.

Características del problema como la existencia de un flujo de actividades, su duración en el tiempo o la necesidad de persistencia, se prestan a la utilización de un sistema de gestión de procesos de negocio, BPMS, lo que permitiría la

gestión del ciclo de vida de una subvención de forma ágil y eficiente, y además reutilizar la lógica de negocio ya existente.

Por ello la empresa Sinergia Tecnológica decidió abordar el proyecto piloto mencionado anteriormente, en el que el caso de estudio se ha centrado en cubrir toda la primera fase del ciclo de vida de una subvención, con la posibilidad de ser extendido a todo el ciclo de vida. Los hitos principales de esta fase son: 1) Registro de la entidad, 2) Registro de curso, 3) Programación del curso, 4) Homologación del centro o Autorización de la entidad, y 5) Baremación del expediente.

Puesto que la lógica de negocio de esta fase se encontraba ya implementada en PL/SQL, se planteó como objetivo aprovechar las capacidades de integración que ofrecen los BPMS para incorporar la lógica de negocio existente.

3. La solución BPM

Business Process Management (BPM) es un nuevo paradigma para crear aplicaciones de gestión empresarial centradas en el modelado, ejecución, administración y monitorización de los procesos de negocio. En los últimos años ha aumentado de forma considerable el interés de las empresas por BPM [1], normalmente complementado con la utilización de una arquitectura orientada a servicios (SOA), y han emergido diversas tecnologías y estándares.

En este apartado, se identifica la estrategia BPM seleccionada, se presenta un esquema genérico de proceso de desarrollo BPM y se describe en detalle como se ha trasladado al caso de BPMS elegido.

3.1. Estándares y herramientas

En torno a BPM se han definido numerosos estándares como son: BPMN, BPEL, XPD, WS-CDL, y un largo etcétera. Son pocos los que se han impuesto (entre ellos los antes mencionados), mientras que la gran mayoría (como BPML o WSFL) han quedado en desuso. Con el objetivo de desarrollar una solución BPM para el problema presentado en el apartado anterior, se ha optado por utilizar el binomio BPMN-BPEL tal y como recomienda el BPMI (*Business Process Management Initiative*).

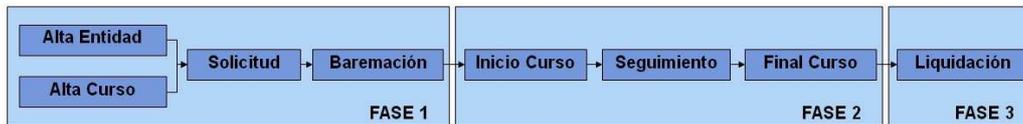


Figura 1. Ciclo de vida de una subvención

BPMN (*Business Process Modeling Notation*) [3] es una notación gráfica para modelar flujos de proceso de negocio, cuya base formal son las redes de Petri. Para el modelado de los procesos de negocio se dispone de otras notaciones, como por ejemplo, los diagramas de actividad de UML 2.0. Estos diagramas y BPMN son muy similares, por lo que es posible que converjan en el futuro. No obstante, como se señala en [4] hay aspectos que UML no soporta pero sí BPMN, como la implantación de SOA.

Por otra parte, BPEL (*Business Process Execution Language for Web Services*) [5] es un lenguaje de ejecución de procesos de negocios basado en XML promovido por OASIS. Está ampliamente difundido y es soportado por la mayoría de herramientas como lenguaje de ejecución. BPEL está inspirado en XLANG y WSFL, a su vez basados en el cálculo Pi y las redes de Petri, respectivamente. BPEL es un lenguaje centrado en el proceso (*process-centric*), y como tal presenta el problema de que no tiene en consideración las interfaces humanas. Como respuesta a esta carencia, ha surgido una extensión de IBM y SAP denominada *BPEL for People* (BPEL4People) [6], que suple la integración con interfaces humanas, pero que no forma parte del estándar de OASIS. Existen lenguajes de ejecución centrados en la interfaz de usuario (*user-centric*), como XPDL, que suponen alternativas a BPEL4People.

Habitualmente, los lenguajes de ejecución permiten la combinación de servicios. Dicha combinación puede ser de dos formas: coreografía u orquestación. En el primer caso, un conjunto de servicios son coordinados por otro que controla las interacciones, de modo que los servicios coordinados no tienen conocimiento de que forman parte de un sistema. En el segundo caso, no existe un coordinador centralizado, sino que cada servicio del sistema colabora con el resto y conoce con quién puede interactuar, cuándo, y cómo realizarlo. BPEL define dos tipos de procesos para soportar la combinación de

servicios: procesos de tipo concreto para soportar la orquestación y procesos de tipo abstracto para la coreografía. WS-CDL es un lenguaje basado en XML, propuesto por el *World Wide Web Consortium* (W3C) para modelar la interacción global de participantes autónomos en un modelo de procesos de negocio. WS-CDL es el estándar más extendido para definir coreografías de servicios.

Los sistemas BPMS permiten el modelado, despliegue y ejecución de los procesos de negocio. Estos sistemas engloban diversas herramientas que se pueden clasificar en tres categorías fundamentales: herramientas de modelado de procesos (mediante BPMN u otro), motores de ejecución (ejecutan código BPEL, XPDL,...), y herramientas de simulación, monitorización y optimización de procesos. Frecuentemente, la herramienta de modelado del BPMS permite la generación automática del código ejecutable (BPEL, etc.), de modo que simplemente se requiere su despliegue en el motor. Un BPMS es una plataforma para representar cualquier modelo de procesos, de modo que workflows, EAI (*Enterprise Application Integration*), o colaboraciones B2B son ejemplos de aplicación.

En el proyecto se ha elegido el BPMS de Intalio [7] ya que soporta el binomio BPMN-BPEL y es una herramienta de código abierto madura. Intalio ofrece un framework con herramientas maduras que permiten modelar procesos de negocio mediante BPMN. A partir de los modelos, la generación a código BPEL es automática. También permite un despliegue automático en el sistema gestor.

3.2. Proceso de desarrollo genérico

En este apartado se describirá de forma breve un esquema de proceso de desarrollo genérico para BPM, que se sustenta en el uso de los estándares más destacados. Este proceso es el propuesto en [8], que a su vez está basado en el modelo de

referencia del WfMC, quizás el más maduro de los diversos grupos de estándares BPM. El esquema del proceso de desarrollo propuesto es el siguiente:

1. **Desarrollar las interfaces requeridas**, tanto del sistema (internas y externas) como humanas. Se parte de los servicios externos (normalmente servicios web) ya implementados, y esta fase se centra en definir interfaces para dichos servicios. También se implementan los formularios web (u otro tipo de interfaz gráfica) en el caso de interfaces humanas, que permiten a los participantes humanos ver y ejecutar las actividades manuales pendientes.
2. **Generar un modelo BPMN inicial** a partir de una coreografía WS-CDL. Esta fase debe omitirse si el proceso no deriva de una coreografía. Para aplicaciones que involucran interacciones complejas con participantes externos (por ejemplo, en colaboraciones B2B), las herramientas de coreografía WS-CDL generan un modelo BPMN básico que captura las comunicaciones requeridas del proceso local. También pueden realizar una validación, o comprobación de conformidad de coreografía de ese modelo.
3. **Diseñar el modelo BPMN**. Los analistas de negocio y los desarrolladores realizan un modelo con el lenguaje BPMN mediante herramientas visuales.
4. **Generar el código BPEL** en base al modelo BPMN. Los editores de BPMN suelen incluir una herramienta de generación de código BPEL.
5. **Desplegar el código BPEL y las interfaces** en el motor de ejecución del BPMS por parte de los analistas de negocio y los administradores del sistema.
6. **Monitorizar los procesos** en ejecución mediante las interfaces de administración y monitorización. Los administradores de BPMS usan consolas de monitorización y administración gráfica, con el fin de mantener y rastrear el estado de los procesos del motor de ejecución. La consola utiliza un lenguaje de gestión para interactuar con el motor.

3.3. Proceso de desarrollo en el caso de estudio

A continuación se detallará cómo se ha concretado el proceso genérico anterior para utilizar un BPMS como Intalio.

Primero se han desarrollado las interfaces de servicios externos: los servicios web. Estos servicios se generan automáticamente a partir de unos *wrapper* a código PL/SQL, mediante la aplicación de técnicas MDD. En esta fase también es necesario desarrollar las interfaces humanas. Intalio recurre al framework Orbeon [9], lo que permite el diseño de formularios web (*XForm*) que se despliegan de forma independiente como servicios web, y pueden ser integrados con el proceso de negocio. Se generan automáticamente esquemas XML y descriptores WSDL a partir de los formularios web, que posteriormente el desarrollador utilizará según las pretensiones con la que se creó del formulario.

El BPMS de Intalio utiliza una notación BPMN extendida con el fin de generar automáticamente el código BPEL. Por este motivo, se ha creído conveniente construir un modelo de proceso de negocio abstracto, esto es, un modelo del proceso de negocio expresado en BPMN y no ligado a notaciones particulares de Intalio, que se añadirán más tarde. La motivación principal para crear este modelo abstracto es proporcionar independencia de las herramientas y mejorar la legibilidad, la cual se reduce al trabajar con un sistema concreto. De hecho, podría realizarse con otra herramienta de modelado BPMN, pero es recomendable utilizar la de Intalio, con el fin de aprovechar parte del diseño. La Figura 2 muestra el modelo BPMN abstracto creado para el caso de estudio. Dado que no se parte de una coreografía WS-CDL, este modelo inicial no se obtiene a partir de ésta, sino que se diseña a partir de los requisitos. El modelo BPMN que se ha realizado abarca desde la solicitud de cursos hasta su aprobación.

El siguiente paso consiste en transformar el modelo BPMN abstracto en un modelo concreto, considerando las extensiones de la herramienta utilizada. Evidentemente, cuanto más se ajuste el lenguaje de modelado de la herramienta al estándar BPMN más sencilla será la adaptación. En Intalio, la adaptación del modelo abstracto

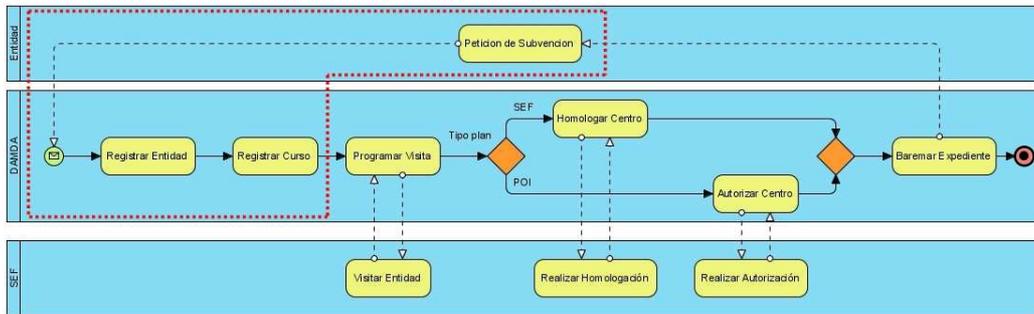


Figura 2. Modelo BPMN abstracto

requiere la creación de *pools* para cada interacción con un participante, inclusión de actividades correlacionadas, anotación de flujos, y especificación de las operaciones BPEL en las actividades.

- Se deben crear *pools* para cada interacción con un participante. Aunque se trate del mismo participante, es necesario crear un *pool* diferente para cada tarea que interactúe con un participante humano (una limitación de la versión 4.4.1 de Intalio, que se corregirá en posteriores versiones).
- Se deben incluir actividades adicionales, correlacionadas para las tareas humanas. Se debe subrayar que BPEL es un estándar centrado en el proceso (*process-centric*), no centrado en las tareas de usuario (*user-centric*). La extensión BPEL4People se propuso en el 2005, y por tanto posterior a numerosas herramientas de BPM. Los sistemas BPM que ejecutaban BPEL necesitaban manejar interfaces humanas, de modo que ante las deficiencias del estándar, se ingeniaron extensiones propietarias. Éste es el caso de la solución tecnológica de Intalio. En Intalio, una tarea permite mostrar una información a determinado usuario, y que éste introduzca manualmente determinados datos. Entre el inicio de la tarea y la finalización de la misma pueden transcurrir días, pero otras ramas del proceso que son independientes pueden continuar su ejecución. Intalio contempla esta situación mediante dos actividades en el proceso: una representa el inicio de la tarea y otra la finalización de la misma. La correlación de ambas actividades es una tarea necesaria para ligar dichas

actividades, que se realiza mediante las facilidades de Intalio. En la Figura 3 se ilustra cómo se adapta la sección punteada de la Figura 2. Se puede apreciar la inclusión de una actividad para el inicio de la tarea, y otra para la finalización, tal y como se ha comentado.

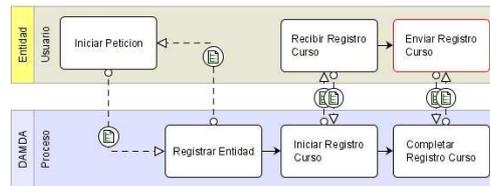


Figura 3. Sección del modelo BPMN concreto

- Es necesario anotar los flujos entre *pools*. El objetivo de esta tarea es ligar los flujos con operaciones del WSDL, y concretar de este modo qué cantidad y qué tipos de datos envía o recibe cada actividad. El entorno permite realizar esta tarea de forma visual, obteniendo esta información de los WSDL generados a partir de los XForm o de los servicios web externos. Los esquemas XML difieren según el propósito para el que se haya diseñado el formulario, pues dependiendo de si se trate de una notificación, la creación de una tarea, o un proceso iniciado por un usuario (PIPA), los tipos de datos adicionales de gestión utilizados por la herramienta son diferentes. En la Figura 3 se observa un icono en los flujos, indicando que tienen asignada una operación de un WSDL.

- Se requiere especificar las operaciones BPEL que realizan las tareas. Intalio facilita un editor gráfico de correspondencias (*mappings*) entre los mensajes de los flujos, de modo que para cada actividad, se indica qué operaciones se llevan a cabo entre los datos, y qué intercambios de datos existen con otras actividades.

A continuación se genera el código BPEL. Este paso se realiza de forma automática a partir del modelo BPMN extendido y la información de *mappings*. Una vez se dispone del código BPEL, se despliega junto a las interfaces en el BPMS. El despliegue del código BPEL y los formularios (*XForm*) al sistema Intalio también se realiza de forma automática. En el caso de que un proceso sea iniciado por un usuario (PIPA), es necesario para su despliegue recurrir a otra herramienta de Intalio que se distribuye de forma independiente a la herramienta de modelado del BPMS. Los servicios web para los servicios externos se despliegan de forma independiente al framework de Intalio.

La última etapa es la monitorización de procesos. El sistema gestor Intalio incluye facilidades para la gestión y monitorización de los procesos de negocio que están desplegados. La aplicación web de gestión permite acciones típicas como el arranque, parada y eliminación de despliegues de los procesos. La aplicación de monitorización incluye una utilidad consistente en un diagrama BPMN del proceso en el que se indica qué actividades del proceso se encuentran en ejecución, y en qué estado, así como las trazas de los errores ocasionados en el servidor.

3.4. Integración de Servicios Web

Cuando se implementa un flujo de procesos de negocio con un BPMS es común reutilizar la lógica de negocio existente en la organización. Para ello, la lógica de negocio es encapsulada en servicios, los cuales son utilizados por las actividades de los flujos de proceso de negocio. Con el uso de BPEL, estos servicios se implementan con el uso de servicios web.

Con la intención de integrar algunas funciones de la lógica de negocio existente en el flujo de procesos de negocio expuesto anteriormente, se optó por una solución basada en MDD para la

generación automática de servicios web. Los servicios web generados acceden a la lógica de negocio a través de los *wrappers* Java mencionados en el apartado 2. La generación automática de estos servicios web se ha ideado como una transformación modelo-código, que parte de un modelo PL/SQL obtenido con la técnica explicada en [2]. De esta forma se incorpora un tercer nivel de abstracción para acceder al código PL/SQL.

Un aspecto muy importante tenido en cuenta es la seguridad. Para incorporar la seguridad a los servicios web generados se hizo uso de la especificación WS-Security. Como una primera aproximación, se utilizó el modo básico de nombres de usuario y huella digital en el paquete SOAP. Las opciones de seguridad se pueden especificar por medio de anotaciones en la especificación PL/SQL.

La incorporación de los servicios web generados automáticamente al flujo de procesos de negocio se realiza a través del fichero WSDL. En la plataforma Intalio, incorporar servicios web externos (definidos externamente a Intalio) consiste, por una parte, en anotar el diagrama BPMN con las operaciones definidas en el fichero WSDL y, por otra parte, desplegar los servicios web en el sistema gestor de procesos de negocio.

4. Valoración

En este apartado se comentan algunas de las ideas extraídas del caso de estudio y que sirven para responder a algunas de las cuestiones comúnmente planteadas en torno al paradigma orientado a procesos soportado por los BPMS (que se referenciará como paradigma BPM). En primer lugar, consideramos que se trata de un paradigma de Desarrollo Dirigido por Modelos (MDD), en el que los modelos de alto nivel del proceso de negocio expresados en BPMN son traducidos a código BPEL ejecutable por un intérprete. Por tanto, BPMN es un lenguaje de modelado gráfico específico del dominio de procesos de negocio y destinado a los analistas de negocio y desarrolladores, mientras que BPEL es otro lenguaje específico del dominio de procesos de negocio orientado a la ejecución por la máquina a través de un motor BPEL. Tanto uno como otro tienen una base formal.

BPMN permite construir modelos de procesos de negocio independientes de la computación (CIM en la terminología de MDA), aunque los modelos necesarios para generar el código BPEL que se despliega en un BPMS contienen información específica de la plataforma de despliegue, por lo que no son independientes de la computación ni de la plataforma. En el contexto de MDA, aunque un modelo BPMN (al igual que un diagrama de actividades de UML) puede ser utilizado como un modelo CIM, surge el problema de transformar dicho modelo en un PIM (diagramas de clase o interacción de UML), lo cual es una tarea compleja que no es soportada por las herramientas MDA en la actualidad, y quizás tampoco en el futuro [10].

Aunque hay autores como [4] que han relacionado el paradigma BPM con MDA, son paradigmas claramente diferentes y con objetivos dispares como se señala en [11]. En MDA, el objetivo es separar la especificación de la funcionalidad de un sistema de su implementación sobre una determinada plataforma, con el fin de favorecer la interoperabilidad. El paradigma BPM está orientado a la automatización de procesos de negocio a partir de un esquema expresado en BPMN, aunque cabe destacar que también cubre áreas como por ejemplo el modelado e integración.

No obstante, el paradigma BPM puede beneficiarse si se basa en los estándares de metamodelado definidos para MDA. Así lo entendieron las organizaciones BPMI y OMG cuando decidieron unir sus esfuerzos en el dominio de BPM y definir varios metamodelos estándar, entre los que destaca el metamodelo para definir procesos de negocio (BPDM) y el metamodelo de reglas de negocio (*Semantics of Business Vocabulary and Business Rules*, SBVR). BPMN se ajusta a BPDM, el cual está definido en MOF, con lo que se obtendrán beneficios como unificación en la representación, intercambio e integración de modelos y facilidad en la definición de transformaciones.

El BPMS utilizado en el caso de estudio utiliza una aproximación basada en técnicas de MDD para la generación de código BPEL a partir de BPMN. Intalio se ajusta adecuadamente al uso de BPMN-BPEL y genera prácticamente todo el código necesario para la ejecución de los procesos de negocio, exceptuando los servicios externos.

Además, la orquestación de servicios web generada se ajusta a una arquitectura SOA. Resuelve las carencias de BPEL, relacionadas con los aspectos de *user-centric*, utilizando la solución de Orbeon, aunque ello implica particularizar el diagrama BPMN. Aunque es una herramienta con una cierta madurez (creada en el 2000), todavía no está exenta de errores y la curva de aprendizaje es muy pronunciada.

Durante el desarrollo, el modelado del proceso de negocio se realizó utilizando dos modelos: el modelo abstracto y el modelo concreto. Esta decisión de diseño fue tomada porque Intalio utiliza una notación BPMN extendida para suplir las carencias de BPMN desde el punto de vista de la generación automática del código BPEL. Por ejemplo, a partir de un modelo BPMN puro, los mensajes entre actividades no contienen información suficiente para poder inferir correctamente el código BPEL. Las particularidades introducidas por Intalio para permitir las interacciones humanas reducen la legibilidad y portabilidad del modelo BPMN, de ahí el interés del modelo abstracto.

5. Conclusiones

Tal y como se señala en [12] hay una carencia de buenos casos de estudio sobre BPM. En este trabajo se ha presentado un caso de estudio de utilización de un BPMS en el ámbito de la Administración Pública. El caso de estudio forma parte de un proyecto piloto que, a falta de completar la fase de pruebas y evaluación, se ha podido comprobar una mejora en la productividad al utilizar BPMS y la generación de servicios web aplicando MDD.

Como trabajo futuro, se valorará el impacto que produce la solución BPM en la empresa teniendo en cuenta aspectos como la productividad y la adaptación al cambio. Por otra parte, se profundizará en la relación de BPM con MDD, con el objetivo de analizar los beneficios de su aplicación conjunta.

Agradecimientos

Este trabajo ha sido parcialmente financiado por los proyectos 2I05SU0018 y TIC-INF 06/01-0001 de la Consejería de Educación y Cultura (CARM).

Referencias

- [1] *Key issues for BPM*, Gartner, Marzo, 2007.
- [2] J.L. Cánovas, O. Sánchez, J. Sánchez, J. García, *Utilidad de las transformaciones modelo-modelo en la generación automática de código*, XII Jornadas de Ingeniería del Software y Bases de Datos, CEDI 2007. Pendiente de publicación.
- [3] BPMN Specification, <http://www.bpmn.org/Documents>, 2006.
- [4] P. Harmon, *The OMG's Model Driven Architecture and BPM*, BPTrends, 2004.
- [5] BPEL Specification, <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.pdf>, 2006.
- [6] BPEL4People Specification, <ftp://www6.software.ibm.com/software/developer/library/wsbpel4people.pdf>, 2005.
- [7] Intalio, <http://www.intalio.com>.
- [8] M. Havey, *Essential Business Process Modeling*, O'Reilly, 2005.
- [9] Orbeon, <http://www.orbeon.com>.
- [10] *BPM and MDA: The Rise of Model-Driven Enterprise Systems*, BPTrends, June, 2003.
- [11] H. Smith, *BPM and MDA: Competitors, Alternative or Complementary*, BPTrends, 2003.
- [12] M. Z. Muehlen, *Class Notes: BPM research and Education*, BPTrends, 2007.