

Hacia la Obtención de Procesos de Negocio desde Sistemas de Información Heredados

Alfonso Rodríguez¹, Angélica Caro¹

¹Departamento de Ciencias de la Computación y Tecnologías de la Información

Universidad del Bío-Bío

Casilla 447, Chillán

Chile

{alfonso,mcaro}@ubiobio.cl

Resumen. En la actualidad las empresas están prestando mucha atención a sus procesos de negocio, ya que reconocen que ellos son un valioso recurso con el cual es posible obtener ventajas competitivas en el mercado. Además, han aparecido nuevas notaciones y lenguajes que permiten representar los procesos de negocio, facilitando con ello su identificación y entendimiento. Al mismo tiempo, las organizaciones conviven con sistemas de información heredados que resultan ser fundamentales para su desempeño. Estos sistemas, además de su dimensión técnica, contienen información acerca de la estructura organizacional, la estrategia, los procesos y los flujos de trabajo. No obstante, esta información, por lo general, no se encuentra directamente disponible para los analistas de negocio. Una forma de visualizar esta información es como procesos de negocio. En este artículo discutimos los primeros avances respecto de la definición de un marco de trabajo que, bajo un enfoque dirigido por modelos, permitirá obtener procesos de negocio desde los sistemas de información heredados.

1. Introducción

Hoy en día los procesos de negocio (BP, siglas inglesas de Business Process) están recibiendo mucha atención por parte de la comunidad científica y las empresas, ya que se ha entendido que son un recurso importante para el desempeño y la mantención de la competitividad en este tipo de organizaciones. Por lo mismo, una necesidad fundamental que se deriva de esto es la visualización de los BP. Esto significa poder contar

con una descripción de los BP en un lenguaje que permita tener los modelos disponibles para su entendimiento, adaptación y mejora. Atendiendo a esta necesidad, durante la última década, aparece la notación BPMN (Business Process Modeling Notation) [1] y el lenguaje UML (Unified Modeling Language) [2] es mejorado para permitir la representación de procesos de negocio. Ambos, la notación y el lenguaje, constituyen una valiosa herramienta para la descripción de procesos de negocio puesto que cumplen un doble propósito: (1) facilitar la labor de los analistas de negocio (lenguaje gráfico) y (2) servir de punto de partida en un proceso de desarrollo de software para los analistas de sistemas [3].

Por otro lado, las empresas tienen un conjunto de sistemas de información que les permiten llevar a cabo una buena parte de las tareas que apoyan su desempeño en el mercado. Un número importante de esos sistemas puede ser clasificado en la categoría de sistemas de información heredados (LIS, siglas inglesas de Legacy Information Systems). Estos sistemas tienen mucho más que una dimensión técnica, ellos abarcan elementos de la estructura organizacional, la estrategia, los procesos y los flujos de trabajo de la empresa. Un componente importante de los LIS es el software heredado. El software heredado es imposible de modificar y mantener a costos razonables. Por lo general, el costo de reemplazo excede el costo de seguir operando con él. De manera que las empresas asumen la convivencia con los sistemas y el software heredado como parte de la operación rutinaria, aunque, evidentemente, están conscientes que ello representa un problema. Justamente, un problema es que los BP asociados a ellos no siempre están accesibles (visibles) mediante un modelo y en los casos en que sí lo están, pueden no estar actualizados [4].

El trabajo presentado en este artículo considera este escenario y apunta a obtener el máximo de beneficios desde los LIS usándolos para visualizar (describir) procesos de negocio con los que se vinculan. Para ello se han identificado los elementos básicos que constituyen una primera aproximación al marco de trabajo que se propone.

El objetivo final es desarrollar un marco de trabajo (llamado LIS2BP) que permita obtener a partir de un LIS, de manera semi-automática, el o los BP asociados a él. Para esto, se ha considerado utilizar la parte *visible* del

software heredado, esto es, programas (fuentes y/o ejecutables), archivos de datos, entradas (descripción de pantallas) y salidas (descripción de pantallas y/o informes), usuarios y secuencias de ejecución. A partir de estos elementos se podrán extraer un BP descrito con UML y/o BPMN (quedarán visibles), con el consecuente beneficio que esto puede tener para la empresa, que podrá modificar y/o mejorar sus BP. Finalmente, esta propuesta está basada en un enfoque dirigido por modelos (MD, Model Driven) que en lo esencial considera la transformación de modelos desde y hacia diferentes niveles de abstracción.

El resto de este artículo se organiza de la siguiente manera. La sección 2 describe los conceptos básicos en los que se sustentará el marco de trabajo propuesto. La sección 3 mostrará los avances respecto del desarrollo del marco de trabajo. Y por último, la sección 4 muestra las conclusiones y trabajos futuros.

2. Conceptos básicos

En esta sección se entrega una descripción de los conceptos básicos en los que se sustentará la propuesta del marco de trabajo LIS2BP.

Los procesos de negocio y los sistemas de información heredados

El modelado de procesos de negocio adquiere cada vez más importancia para las empresas y organizaciones. Esto, porque permite registrar, mediante alguna notación, la esencia del negocio y hace posible que diversos actores dentro de una empresa lo entiendan y modifiquen con el propósito de incorporar mejoras. En la actualidad las notaciones más utilizadas para la representación de BP son el Diagrama de Actividad de UML 2.0 (abreviado como UML 2.0-AD) y el Diagrama de Procesos de Negocio de BPMN (abreviado como BPMN-BPD) [5].

Un proceso de negocio es un conjunto de actividades o procedimientos que cumplen un objetivo específico o metas de más largo alcance, en el contexto de una estructura organizacional, definiendo roles funcionales y relaciones [6]. Lo esencial de un BP es que está vinculado con las empresas y que en conjunto definen la forma en que ellas alcanzan sus

objetivos [7]. Dicho de otra forma, es una parte del negocio donde se describen las funciones del mismo y que involucra recursos que son utilizados, transformados o producidos [8].

De acuerdo con Jennings et al. [9], los procesos de negocio están orientados al mercado. Por lo cual se deben ajustar permanentemente para incorporar mejoras que permitan: incrementar la satisfacción del cliente, mejorar la eficiencia en su operación, incrementar la calidad de los productos, reducir costos y/o encontrar nuevos negocios u oportunidades para cambiar los servicios existentes o introducir nuevos.

Por otro lado, las empresas poseen diversos Sistemas de Información a través de los cuales implementan y/o soportan sus BP. Dada la relación existente, resulta lógico pensar que un cambio en los BP debiera repercutir en los sistemas que lo soportan. Sin embargo esto no siempre ocurre y los BP y sistemas de información evolucionan independientemente [4].

Entre los sistemas de información que posee una empresa, normalmente existen algunos que presentan características muy peculiares. Estos sistemas se denominan Sistemas de Información Heredados, y se definen como cualquier sistema de información que se resiste significativamente a cambios y modificaciones [10]; y que normalmente son de misión crítica dentro de una empresa [11]. Esto último significa que si alguno de ellos falla o se detiene traerá graves consecuencias en el desempeño de la empresa. De acuerdo con Wu et al. [12], este tipo de sistemas conforman la columna vertebral del flujo de información en una organización y son el principal vehículo para la consolidación de información acerca del negocio de ésta. La tabla 1 muestra sus principales características.

Tabla 1. Características de un LIS

Típicamente son grandes, con millones de líneas de código
Son antiguos, más de 10 años desde su construcción
Escritos en un lenguaje heredado (e.j. COBOL, Pascal)
Se basan en archivos planos o bases de datos heredadas
Generalmente funcionan en hardware obsoleto que es lento y caro de mantener,
Son autónomos (independientes de otras aplicaciones),
Generalmente, son difíciles de comprender y no existe documentación suficiente o apropiada acerca de ellos,
Su mantenimiento implica un alto costo
Suelen ser de “misión-crítica” dentro de la organización

Dadas las características de los LIS, cabe suponer que es muy probable que no existan modelos que representen los BP asociados a ellos, y por tanto obtenerlos se transforma en un gran desafío [4].

El enfoque dirigido por modelos

En los últimos años la ingeniería del software está siendo influenciada por la transformación de modelos, ya que con ello se apunta a resolver los problemas de tiempo, costos y calidad asociados a la creación de software. La forma de resolver este problema se enmarca bajo la denominación general de ingeniería dirigida por modelos (MDE, Model-Driven Engineering). MDE es una tendencia que ha ido adquiriendo relevancia en la ingeniería del software y, que como un aspecto fundamental, considera modelos como entidades de primera clase cuyo propósito es su desarrollo, mantenimiento y evolución mediante la realización de transformaciones de modelos [13].

La arquitectura dirigida por modelos (MDA, por las siglas inglesas de Model Driven Architecture) es un marco de trabajo que ha sido definido para el desarrollo de software (Ver Figura 1, columna izquierda). MDA se encuentra en el ámbito de MDE y su principal objetivo es permitir la creación de modelos que son totalmente independientes de la implementación tecnológica. Dado que MDA tiene que ver con modelos, el tratamiento que hace de ellos puede ser de dos maneras [14]:

- *Estandarizando los modelos*: esto se refiere al uso de técnicas que aseguren que todos los modelos usados en el desarrollo del software pueden ser relacionados unos con otros.
- *Usando los modelos*: esto se encuentra en el contexto del desarrollo de software y se refiere a la transición desde modelos abstractos hacia modelos más concretos.

Los modelos de procesos de negocio creados por analistas de negocio, se consideran como modelos independientes de computación [15] (Figura 1, nivel superior de la columna izquierda). Mediante transformaciones de modelos éstos pueden ser convertidos en modelos independientes de plataforma. En este nivel, los usuarios principales son arquitectos o diseñadores de software. Las especificaciones independientes de plataforma pueden ser transformadas en especificaciones para una plataforma específica y finalmente en un componente software (Figura 1, nivel inferior de la columna izquierda).

Para que cada modelo especificado en el enfoque MDA se pueda relacionar con otros, deben estar basados en MOF (Meta Object Facility) y metamodelos, de modo que pueda comunicarse con cualquier otro modelo del tipo MOF-subordinado [14].

En este contexto los lenguajes para la transformación de modelos constituyen un aspecto fundamental. Las transformaciones se deben expresar de manera clara y precisa por lo que es conveniente usar un lenguaje definido para dicho propósito. Entre los más difundidos se encuentran VIATRA (Visual Automated Transformations) [16], ATL (Atlas Transformation Language) [17] y QVT (Query/View/Transformation) [18], entre otros.

Dado lo anterior, un modelo de procesos de negocio, construido por un analista de negocios, puede ser usado en un proceso de construcción de software. Esto porque a partir de éste se pueden obtener importantes requisitos del sistema, punto de partida en todos los procesos de desarrollo de software modernos.

Finalmente, la modernización dirigida por la arquitectura (ADM, Architecture Driven Modernization) es una propuesta de OMG (Object Management Group) orientada a la obtención de nuevas aplicaciones de software a partir del software existente (Ver Figura 1, columna derecha). ADM considera la aplicación de ingeniería inversa sobre la solución existente, agregando nuevas metas y requisitos del negocio orientados a crear y optimizar la nueva solución [19].

ADM es una alternativa para el análisis de las arquitecturas de software existente orientado a mejorar las iniciativas de mantenimiento de código y adicionalmente provee beneficios para la migración desde lenguajes o plataformas obsoletas o envejecidas hacia entornos más modernos. Un aspecto importante de ADM es la transformación [20]. Se establecen tres tipos de transformaciones, (i) la formal que establece la necesidad de contar con la descripción formal de los artefactos iniciales, los cambios que se implementarán y las reglas/procesos que regirán la aplicación de los cambios, (ii) la transformación ampliada incorpora la descripción no formal de artefactos y reglas y (iii) los niveles de abstracción de las transformaciones que son utilizados para la descripción de las reglas y artefactos que se encuentran en distintos niveles de abstracción.

ADM aún se encuentra en pleno desarrollo (ver <http://adm.omg.org/>), pero ha sido mencionado en esta discusión ya que se espera contar con definiciones más precisas que permitan incorporarlo en el desarrollo de este marco de trabajo.

Basados en los conceptos presentados en esta sección, se persigue la definición del marco de trabajo llamado LIS2BP. Tal como se ilustra en la Figura 1 (columna central), LIS2BP se aplicará sobre un LIS (considerando todos los elementos y artefactos relevantes que se posean de él) y mediante un proceso de extracción, análisis y transformaciones se obtendrá el o los procesos de negocio subyacentes al LIS.

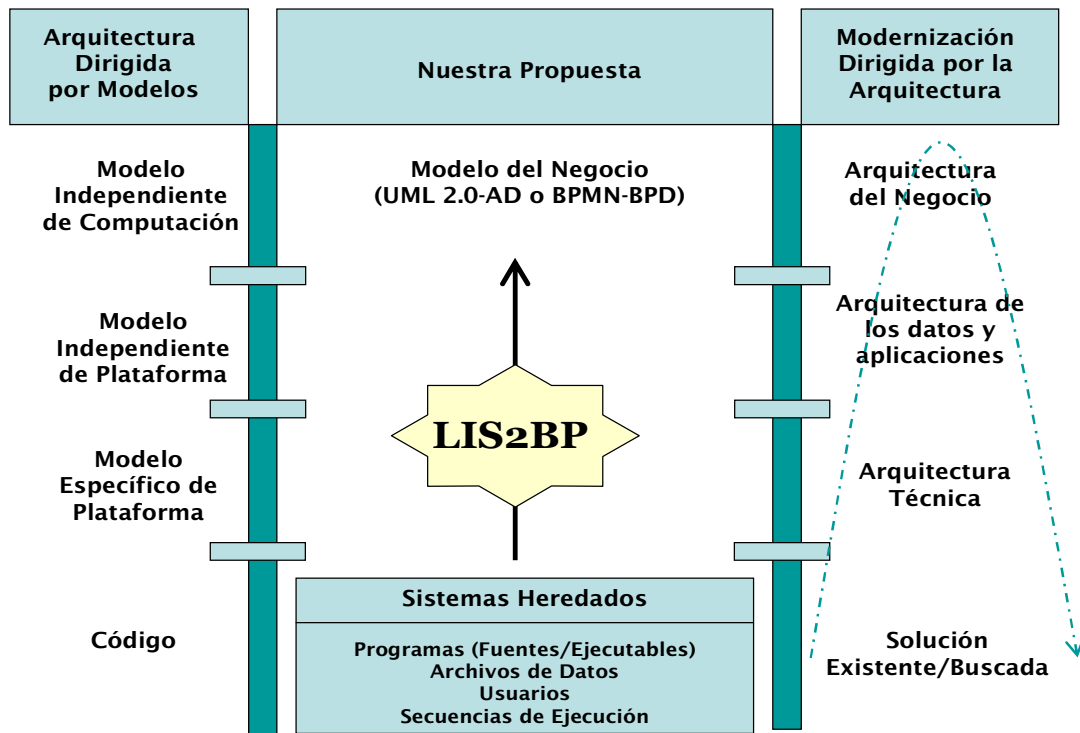


Fig. 1. Marco de Trabajo LIS2BP

En esta propuesta, la arquitectura dirigida por modelos entrega el marco de referencia para la especificación de las transformaciones desde el LIS hacia modelos independientes de computación. Por otra parte, ADM provee el marco de referencia para la aplicación de ingeniería inversa basada en la arquitectura existente, orientada a lograr definiciones del negocio.

3. Hacia la definición de LIS2BP

La esencia del marco de trabajo LIS2BP la constituye la transformación de la información obtenida desde el sistema de información heredado en uno o más modelos de procesos de negocio. Para ello es necesario definir un conjunto de reglas mediante las cuales se establezcan en forma precisa las transformaciones. Estas reglas implementarán un conjunto de heurísticas que serán obtenidas a partir de datos reales disponibles, que consideran más de 20 sistemas heredados que en conjunto contemplan más de 380 archivos de datos y sobre 1.600 programas fuentes.

Para la definición de las reglas se ha considerado la equivalencia entre los elementos componentes del LIS y los elementos que componen un proceso de negocio. En la Tabla 2 se muestra el detalle de estas equivalencias.

La información del sistema heredado, primeras dos columnas de la Tabla 2, ha sido clasificada en: programas fuentes (LIS_SC), programas ejecutables (LIS_XC), archivos de datos (LIS_DF), información acerca de los usuarios del sistema (LIS_US) e información acerca de la periodicidad de ejecución de las tareas más relevantes del sistema heredado (LIS_TM). En la columna central se indica el tipo de elemento del proceso de negocio al cual es equivalente cada uno de los elementos identificados en el LIS. Finalmente, en las dos últimas columnas se muestra la equivalencia con los elementos de UML 2.0-AD y/o BPMN-BPD.

Tabla 2. Equivalencia entre elementos del LIS y BP

Información del sistema heredado		Equivalencia Proceso de Negocio	Representación BP	
Nomenclatura	Significado		UML 2.0-AD	BPMN-BPD
LIS_SC	Programas fuentes			
LIS_SC01	Archivos (claves)	Almacen de Datos	DataStoreNode	Data Object
LIS_SC02	Llamadas a programas	Secuencia de ejecución		
LIS_SC03	Entradas	Documentos	DataStoreNode	Data Object
LIS_SC04	Salidas	Documentos	DataStoreNode	Data Object
LIS_XC	Programas ejecutables			
LIS_XC01	Interfaces			
LIS_XC01_1	Roles	Participantes	ActivityPartition	Pool/Lane
LIS_XC01_2	Secuencia de ejecución	Secuencia Actividades	ObjectFlow	Sequence Flow
LIS_XC01_3	Entradas	Documentos	DataStoreNode	Data Object
LIS_XC01_4	Salidas	Documentos	DataStoreNode	Data Object
LIS_DF	Archivos de datos			
LIS_DF01	Reglas de negocio	Actividades	Actions	Activities
LIS_US	Usuarios			
LIS_US01	Roles	Participantes	ActivityPartition	Pool/Lane
LIS_US02	Distribución Geográfica	Comunicación participantes		Message Flow
LIS_TM	Tiempo			
LIS_TM01	Periodicidad de ejecución	Condiciones temporales		StartEventTimer

Para especificar las transformaciones se usará el lenguaje QVT. Este lenguaje permite especificar en forma clara y no ambigua la equivalencia entre ambos dominios. Un ejemplo de una regla especificada en QVT textual en que se transforma desde LIS hacia UML 2.0-AD se muestra en la Tabla 3. En esta regla se establece que un archivo de datos del sistema heredado obliga a crear un almacén de datos, DataStoreNode en UML 2.0-AD, que tendrá el mismo nombre.

Tabla 3. Regla de transformación en QVT textual

```

transformation LIS2UML-AD
top relation R1
{
  checkonly domain
  lis_file ff:flatfile {name=n}
  enforce domain
  uml_ActivityDiagram dsn:DataStoreNode
  {name=n}
}
  
```

Para la aplicación ordenada y sistemática del conjunto de reglas de transformación se requiere un método que permita llevar a cabo esta tarea. Para ello se ha definido, en forma preliminar, el método M-LIS2BP. Este método comprende un conjunto de etapas, trabajadores, herramientas y artefactos que, bajo un enfoque de ingeniería y exento de ambigüedad, permite crear procesos de negocio a partir de la información disponible en sistemas de información heredados. M-LIS2BP está compuesto por cuatro etapas, tres tipos de trabajadores, dos tipos de herramientas y tres artefactos como se muestra en la Figura 2.




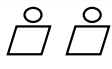






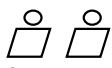


ETAPAS	TRABAJADORES	HERRAMIENTAS	ARTEFACTOS
EXTRACCIÓN		 Programas Ad-Hoc	 Información del LIS  Repositorio LIS
REFINAMIENTO INFORMACION SISTEMA HEREDADO	 Analista Sistemas Operador Sistemas	 Programas Ad-Hoc	 Información del LIS  Repositorio LIS (actualizado)
CREACIÓN PROCESO NEGOCIO		 Reglas (QVT) UML 2.0-AD/BPMN-BPD	 Repositorio LIS  Proceso Negocio
REFINAMIENTO PROCESO NEGOCIO	 Analista Negocio Analista Sistemas	 UML 2.0-AD BPMN-BPD	 Proceso de Negocio (actualizado)

Fig. 2. El método M-LIS2BP

En la primera etapa, llamada EXTRACCIÓN, las tareas se realizan en forma automática por lo que no se requiere la intervención de trabajadores. Las herramientas que se utilizan en esta etapa corresponden a un conjunto de

programas diseñados ad-hoc los que permiten extraer la información disponible en el sistema de información heredado. Estos programas actúan sobre archivos residentes en el computador desde los cuales se obtiene, por ejemplo, los nombres de programas, los nombres y las estructuras de archivos, las secuencias de ejecución de programas (si es que se cuenta con los programas fuentes), los formatos de entradas y salidas, entre otros. Con esta información se hace un llenado inicial del repositorio LIS, el artefacto generado en esta etapa, el cual contendrá la información extraída en forma automática. Tanto el Repositorio LIS como el tratamiento del mismo se encuentran fuera del ámbito de este artículo por lo que sólo se hará referencia al mismo como un artefacto que será creado y actualizado en el contexto de M-LIS2BP.

En la etapa de REFINAMIENTO DEL SISTEMA HEREDADO, la principal tarea corresponde a los trabajadores analista de sistemas y operador de sistemas. Ellos, en su calidad de usuarios conocedores del sistema, tienen la responsabilidad de completar la información acerca del sistema heredado. Esta información es relevante, principalmente, en sistemas en los que no se cuenta con los programas fuentes. La información provista por los trabajadores servirá para actualizar el repositorio.

La tercera etapa, denominada CREACIÓN DEL PROCESO DE NEGOCIO, se lleva a cabo de manera automática. En esta etapa se aplican el conjunto de reglas de transformación definidas en que se establece la equivalencia entre los diferentes elementos del modelo LIS y del modelo de negocios. Las transformaciones deben ser descritas usando QVT (ver ejemplo en Tabla 3) y tendrán que implementar la reglas que han sido descritas en la Tabla 2. Las herramientas de esta etapa son las reglas QVT y los lenguajes para el modelado de procesos de negocio UML/BPMN. Los artefactos utilizados en esta etapa son el repositorio LIS que es utilizado como entrada y el modelo de procesos de negocio que es generado en forma automática.

Finalmente, en la cuarta etapa llamada REFINAMIENTO DEL PROCESO DE NEGOCIO, se procede a mejorar la especificación que se ha obtenido automáticamente. El o los modelos de procesos de negocio serán revisados y refinados por los trabajadores que intervienen en esta etapa. Ellos son el analista de negocio y analista de sistemas que en su calidad de

conocedores del negocio y del sistema respectivamente analizarán y completarán el proceso de negocio. La herramienta utilizada en esta etapa es el lenguaje de modelado de proceso de negocio (UML/BPMN) y el artefacto generado es el proceso de negocio mejorado.

4. Conclusiones y Trabajo Futuro

En este artículo, se ha utilizado el enfoque dirigido por modelos para contextualizar los principales elementos que nos permiten ir hacia la definición del marco de trabajo LIS2BP. Este marco de trabajo considera la obtención de procesos de negocio desde la información existente en sistemas de información heredados. Para el desarrollo de éste se cuenta con suficientes datos para la definición de las heurísticas que permitirán establecer la equivalencia entre los elementos del LIS y los BP. Mediante el uso del lenguaje QVT se definirán las reglas que permitirán llevar a cabo las transformaciones. La aplicación ordenada y no ambigua de todos los elementos del marco de trabajo se ha presentado como un método llamado M-LIS2BP.

Consideramos que el marco de trabajo propuesto será de gran utilidad para las organizaciones, ya que a partir de su aplicación éstas podrán: (i) visualizar (describir) los procesos de negocio asociados a los sistemas de información heredados, (ii) disponer de los procesos de negocio para su entendimiento, adaptación y mejora, (iii) mejorar la gestión del negocio; puesto que al contar con las descripciones de los BP obtenidos desde los LIS, éstos podrán ser coordinados de manera conjunta con el resto de los BP de la organización y, finalmente, (iv) propiciar la evolución de los sistemas de información heredados en el marco de la modernización dirigida por la arquitectura (ADM).

El trabajo futuro se centrará en la definición detallada de las heurísticas y de todas las etapas del método. Finalmente, se elaborarán casos de estudio que permitan refinar la propuesta.

Referencias

- [1] BPMN; *Business Process Modeling Notation (BPMN)*. In <http://www.bpmn.org/Documents/BPMN%20V1-0%20May%203%202004.pdf>. (2004).
- [2] Object Management Group; *Unified Modeling Language: Superstructure*, version 2.0, formal/05-07-04. In <http://www.omg.org/docs/formal/05-07-04.pdf>. (2005).
- [3] List, B. and Korherr, B.; *A UML 2 Profile for Business Process Modelling*, 1st International Workshop on Best Practices of UML (BP-UML) at ER-Conference. Klagenfurt, Austria. (2005). pp.85-96.
- [4] Zou, Y., Lau, T. C., Kontogiannis, K., Tong, T. and McKegney, R.; *Model-Driven Business Process Recovery*, 11th Working Conference on Reverse Engineering (WCRE 2004). 8-12 November 2004, Delft, The Netherlands. (2004). pp.224-233.
- [5] Lonjon, A.; *Business Process Modeling and Standardization*, BPTrends. <http://www.bptrends.com/>. (2004).
- [6] WfMC, *Workflow Management Coalition: Terminology & Glossary.*, (1999). p.65.
- [7] Aguilar-Savén, R. S.; *Business process modelling: Review and framework*, International Journal of Production Economics. Vol. 90 (2). (2004). pp.129-149.
- [8] Eriksson, H.-E. and Penker, M., *Business Modeling with UML*, OMG Press. (2001).
- [9] Jennings, N. R., Norman, T. J., Faratin, P., O'Brien, P. and Odgers, B.; *Autonomous Agents for Business Process Management*, Applied Artificial Intelligence. Vol. 14 (2). (2000). pp.145-189.
- [10] Brodie, M. and Stonebraker, M., *Migrating Legacy Systems: gateways, Interfaces and the Incremental Approach.*, (1995). p.
- [11] Bisbal, J., Lawless, D., Richardson, R., Wu, B., Grimson, J., Wade, V. and O'Sullivan, D.; *An Overview of Legacy Information Systems Migration*, APSEC'97/ICSC'97: Joint 1997 Asia Pacific Software Engineering Conference and International Computer Science Conference. Hong Kong, China. (1997). pp.529-530.
- [12] Wu, B., Lawless, D., Bisbal, J., Richardson, R., Grimson, J., Wade, V. and O'Sullivan, D.; *Legacy System Migration: A Legacy Data Migrating Engine*, 17th DATASEM '97. Brno, Czech Republic. (1997). pp.129-138.

- [13] Mens, T. and Van Gorp, P.; *A Taxonomy of Model Transformation*, Electronic Notes in Theoretical Computer Science. Vol. 152. (2006). pp.125-142.
- [14] Harmon, P.; *The OMG's Model Driven Architecture and BPM*, .Business Process Trends, Vol. 2 (5). In <http://www.bptrends.com/publicationfiles/05%2D04%20NL%20MDA%20and%20BPM%2Epdf>. (2004).
- [15] Rodríguez, A., Fernández-Medina, E. and Piattini, M.; *Towards CIM to PIM transformation: from Secure Business Processes defined by BPMN to Use Cases*, 5th International Conference on Business Process Management (BPM). Lecture Notes in Computer Science Volumen 4714. Brisbane, Australia. (2007). pp.408-415.
- [16] Csertan, G., Huszerl, G., Majzik, I., Pap, Z., Pataricza, A. and Varro, D.; *VIATRA-Visual Automated Transformations for Formal Verification and Validation of UML Models*, 17th IEEE International Conference on Automated Software Engineering (ASE). (2002). pp.267-270.
- [17] Bézivin, J., Breton, E., Dupé, G. and Valduriez, P., *The ATL Transformation-based Model Management Framework*, N° 03.08, IRIN-Université de Nantes. (2003). p.17.
- [18] QVT, *Meta Object Facility (MOF) 2.0 Query/View/Transformation Specification*, OMG Adopted Specification ptc/05-11-01, (2005). 204 p.
- [19] Khusidman, V. and Ulrich, W.; *Architecture-Driven Modernization: Transforming the Enterprise*. In www.omg.org/docs/admtf/07-12-01.pdf. (2007).
- [20] Khusidman, V.; *ADM Transformation*. In www.omg.org/adm/ADMTransformationv4.pdf. (2008).