

Enl@ce: Revista Venezolana de Información,  
Tecnología y Conocimiento  
ISSN: 1690-7515  
Depósito legal pp 200402ZU1624  
Año 8: No. 1, Enero-Abril 2011, pp. 11-29

Cómo citar el artículo (Normas APA):  
Losavio, F., Guzmán, J.C. y Matteo, A. (2011). Correspondencia  
Semántica entre los lenguajes BPMN y GRL. *Enl@ce  
Revista Venezolana de Información, Tecnología y  
Conocimiento*, 8 (1), 11-29

## Correspondencia Semántica entre los lenguajes BPMN y GRL<sup>1</sup>

*Francisca Losavio*<sup>2</sup>  
*Jean Carlos Guzmán*<sup>3</sup>  
*Alfredo Matteo*<sup>4</sup>

### Resumen

Un modelo de negocio es un punto inicial para la elaboración del sistema de software; es útil para que el equipo de desarrollo entienda mejor la especificación de los requisitos globales que el futuro sistema de software debe satisfacer. Sin embargo, cómo pasar del modelo de negocio al modelo del sistema tomando en cuenta metas no funcionales en las etapas tempranas del ciclo de vida del software, es aún un problema por resolver. Actualmente existe un consenso en considerar los aspectos no funcionales como clave para construir la arquitectura del sistema de software, pero estos no son considerados en los modelos de negocio, aunque sus reglas puedan implicar aspectos no funcionales

Recibido: 05-10-10    Aceptado: 22-02-11

---

<sup>1</sup> Este trabajo es parcialmente financiado por el FONACIT del Ministerio de Ciencia y Tecnología de Venezuela.

<sup>2</sup> Profesora Titular del Laboratorio MoST (Modelos, Software y Tecnologías), Escuela de Computación, Facultad de Ciencias, Universidad Central de Venezuela. Doctora en Ciencias de la Computación en 1991 y Doctora de 3er. ciclo en 1985 de la Universidad de Paris-Sud, Paris XI, Orsay, Francia. Fundadora de la Opción de Postgrado en Ingeniería de Software y del Centro de Ingeniería de Software y Sistemas (ISYS) de la Escuela de Computación, Facultad de Ciencias, UCV, en 1992. Sus temas de investigación son en arquitectura, calidad, estándares y proceso de desarrollo de software. Correo electrónico : francislosavio@gmail.com

<sup>3</sup> Lic. en Computación. Consultor en Tecnologías de la Información y Comunicación. Master en Gerencia de TIC. Doctorante en Ciencias de la Computación en la Universidad Central de Venezuela. Correo electrónico: jguzman106@gmail.com

<sup>4</sup> Profesor Titular de la Escuela de Computación, Facultad de Ciencias, Universidad Central de Venezuela. Doctor en Ciencias de la Computación de la Universidad Paul Sabatier, Toulouse, Francia. Coordinador del Postgrado en Ciencias de la Computación, Escuela de Computación, Facultad de Ciencias, UCV. Sus temas de investigación son: modelos, arquitecturas y metodologías para el desarrollo de software. Correo electrónico: almatteo@cantv.net

como por ejemplo *usabilidad* y seguridad. El modelado del proceso de negocio se centra en la representación gráfica de actividades mediante lenguajes de modelado, como *BPMN (Business Process Modeling Notation)*, en los cuales no se destacan las metas no funcionales. Por otra parte, lenguajes orientados a metas como *GRL (Goal-oriented Requirements Language)*, describen el sistema de software centrados tanto en metas funcionales como no funcionales, y permiten representar también entidades y elementos de negocio. El objetivo de este trabajo es presentar una correspondencia semántica entre BPMN y GRL, lenguajes conocidos y ampliamente utilizados, y se describen reglas precisas para esta correspondencia, resaltando los aspectos no funcionales, la cual será utilizada para la derivación del modelo arquitectónico inicial del sistema, a partir del modelo de negocio, en etapas tempranas del proceso de desarrollo, y de esta manera reducir la brecha entre los modelos de negocio y del sistema.

**Palabras clave:** *BPMN*, *GRL*, modelo del proceso de negocio, modelo de negocio, metas no funcionales, correspondencia semántica

## **Semantic Correspondence between GRL and BPMN Languages**

### **Abstract**

Software system development should start from a business model expressing the business logic. It provides the development team a better understanding of the requirements specification that the future software system must satisfy. However, how to move from a business model to a system model, highlighting non-functional goals at early stages of the software life cycle, is still an open problem. There is a general agreement to consider non-functional aspects as crucial to construct the software system architecture. The business process modeling focuses on the graphic representation of activities by modeling languages, such as *BPMN (Business Process Modeling Notation)*. However, these languages do not focus on non-functional goals. On the other hand, goal-oriented languages, such as *GRL (Goal-oriented Requirements Language)*, describe the software systems focusing on non-functional goals, allowing also representing business entities and elements. The aim of this paper is to present the semantic correspondence between BPMN and GRL, well known and widely used representative of the above-mentioned languages. This correspondence, described by precise rules and focused on non-functional aspects, will be used to derive the initial architectural model of the system directly from the business model, at early stages of the software development process, reducing the gap between business and system models.

**Key words:** BPMN, GRL, Business Process Model, Business Model, Non-functional goals, Semantic Correspondence

## Introducción

Un *Modelo de Negocio* ofrece una vista abstracta y simplificada de la realidad compleja en la que se expresan conceptos sobre el funcionamiento del negocio de una entidad, en términos de metas y factores de importancia que reflejan su lógica. Está conformado por un diagrama constituido de elementos que contextualizan el negocio, como son actores, actividades, entidades, procesos, vínculos, entre otros (Burlton, 2001). Un *Proceso de Negocio* se concibe como un conjunto vinculado y natural de actividades basadas en habilidades y competencias de trabajo, que incluyen interacciones entre actores, recursos utilizados y vínculos de dependencias entre estos. Las actividades en un proceso de negocio son coordinadas por actores, quienes tienen la *intención* de satisfacer ciertas metas (Crusson, 2006). La especificación de los procesos de negocio permiten al equipo de desarrollo de software, entender de un modo apropiado las necesidades y los requisitos que el futuro sistema debe satisfacer. El *Modelado del Proceso de Negocio* está centrado principalmente en la representación gráfica de esos procesos, usando lenguajes de modelado específicos tales como la Notación de Modelado de Procesos de Negocio o *Business Process Modeling Notation* (BPMN) (BPMIG, 2002) y el Lenguaje de Requisitos orientado a Metas o *Goal-oriented Requirements Language* (GRL) (ITU-T, 2003, ITU-T, 2008). El lenguaje BPMN iniciado por el *Business Process Modeling Initiative Group* y recientemente admitido por el *Object Management Group* (OMG, 2009) como estándar de facto, tiene por objetivo facilitar la

comprensión del proceso de negocio por parte de los participantes (o *stakeholders*). BPMN soporta solo conceptos de modelado aplicables al negocio desde un nivel de abstracción organizacional intermedio, permitiendo definir, documentar, organizar y rediseñar procesos desde una perspectiva particular. No trata las metas no funcionales que, por ejemplo, pueden ser derivadas de las reglas del negocio: declarar el acceso restringido a cierta información para algunos usuarios implica un objetivo de seguridad, que debe ser necesariamente considerado. Por otra parte, el lenguaje *GRL*, iniciado recientemente por la *International Telecommunication Union* (ITU) como subconjunto del estándar z.151 denominado Definición del Lenguaje de la Notación de Requisitos de Usuarios o *User Requirements Notation* (URN) *Language Definition* (ITU-T, 2008), tiene su origen en dos amplios lenguajes de modelado orientado a metas, tales como: *i\** (Yu y Mylopoulos, 1993) y el *NFR Framework* (Chung, 1991). *GRL* es un enfoque de Ingeniería de Metas para la especificación de sistemas de software, que también permite modelar procesos de negocio flexibles; en las fases tempranas, toma en cuenta la gestión sistemática e intencional de metas, tanto funcionales como no funcionales, desde el punto de vista de la noción de la *intencionalidad distribuida*, que hace énfasis en el cumplimiento de metas por los actores. En este contexto, los términos “intencional” e “intencionalidad”, se refieren al cumplimiento de objetivos o metas desde el punto de vista del usuario o actor. Por otra parte, *BPMN* es un lenguaje muy utilizado en la práctica desde un enfoque de Ingeniería de Procesos de Negocio. Sin embargo, los

diagramas derivados de *BPMN* solo enfatizan los requisitos funcionales del software y no facilitan la definición de una arquitectura inicial. En cambio, el lenguaje *GRL* permite expresar metas de alto nivel organizacional, metas del software y metas de los *stakeholders*, explicitar dependencias entre actores, distinguir entre agentes, posiciones y roles (Amyot, Horkoff, Gross y Mussbacher, 2009), y así razonar sobre cada actor con respecto a sus relaciones con otros actores. El objetivo de este trabajo es establecer reglas precisas para una correspondencia semántica entre los lenguajes *BPMN* y *GRL*, la cual será utilizada en etapas tempranas del desarrollo de software, para la derivación de un modelo arquitectónico inicial del sistema, directamente a partir del modelo de negocio, reduciendo así la brecha entre los modelos de negocio y del sistema de software. En la práctica, muchas organizaciones tienen sus procesos de negocio especificados en algún lenguaje de modelado, por ejemplo *BPMN*, el cual es uno de los lenguajes más usados actualmente en la industria. El paso de un diagrama del proceso de negocio de *BPMN* a un modelo de negocio en *GRL* ayudará a tomar en cuenta en las etapas tempranas las metas no funcionales, que se pueden derivar de las metas organizacionales del modelo de negocio y así facilitar la construcción del modelo arquitectural del sistema. Definimos el término *correspondencia semántica* como la simetría o similitud existente entre dos o más elementos conceptuales pertenecientes a dos o más lenguajes de modelado específicos, tales como *BPMN* y *GRL*.

En este trabajo presentamos pasos y reglas de correspondencia semántica entre los lenguajes

*BPMN* y *GRL*. Se consideró el lenguaje *GRL* en lugar de  $I^*$ , porque es un lenguaje orientado a metas más general que engloba a  $I^*$ , ampliando así trabajos anteriores en los cuales se consideraba la correspondencia entre los lenguajes  $i^*$  y *BPMN*, de Cysneiros y Yu (2004), Koliadis, Vranesevic, Bhuiyan, Krishna y Ghose (2007). Así mismo, debemos observar que los enfoques tradicionales de desarrollo de software enfatizan los aspectos funcionales, obviando los aspectos no funcionales, como por ejemplo la seguridad, la confiabilidad y el rendimiento, para etapas posteriores del desarrollo. Esto trae como consecuencia la producción de sistemas difíciles de mantener y de reutilizar. Bajo esta línea de ideas, las reglas de correspondencia propuestas pueden ser utilizadas en un contexto de reingeniería de proceso, como en el caso de estudio aquí expuesto y como marco de referencia para transformar un diagrama del proceso de negocio de *BPMN* a un modelo de *GRL* y viceversa. Utilizamos como caso de estudio un diagrama del proceso de la Gestión de Trámites de Solicitudes (GTS) tomado del área industrial y adelantado por la Dirección del Talento Humano, que involucra a las Unidades Medulares del *Centro Nacional de Tecnologías de Información* (CNTI), en Caracas, Venezuela, como una primera validación de la propuesta.

El presente trabajo está estructurado de la siguiente manera, además de esta introducción, el agradecimiento y las conclusiones. En la sección 2 se presentan concepto y terminología del modelado de procesos de negocio y el lenguaje *BPMN*. En la sección 3 se describe el lenguaje *GRL*. Luego en la sección 4, se presentan los pasos y reglas para

la correspondencia semántica aplicables a los lenguajes *BPMN* y *GRL*. En la sección 5 se discuten los principales trabajos relacionados con el tema. Finalmente en la sección 6, se aplican las reglas de correspondencia a un caso de estudio.

### **Modelado del proceso de negocio y BPMN**

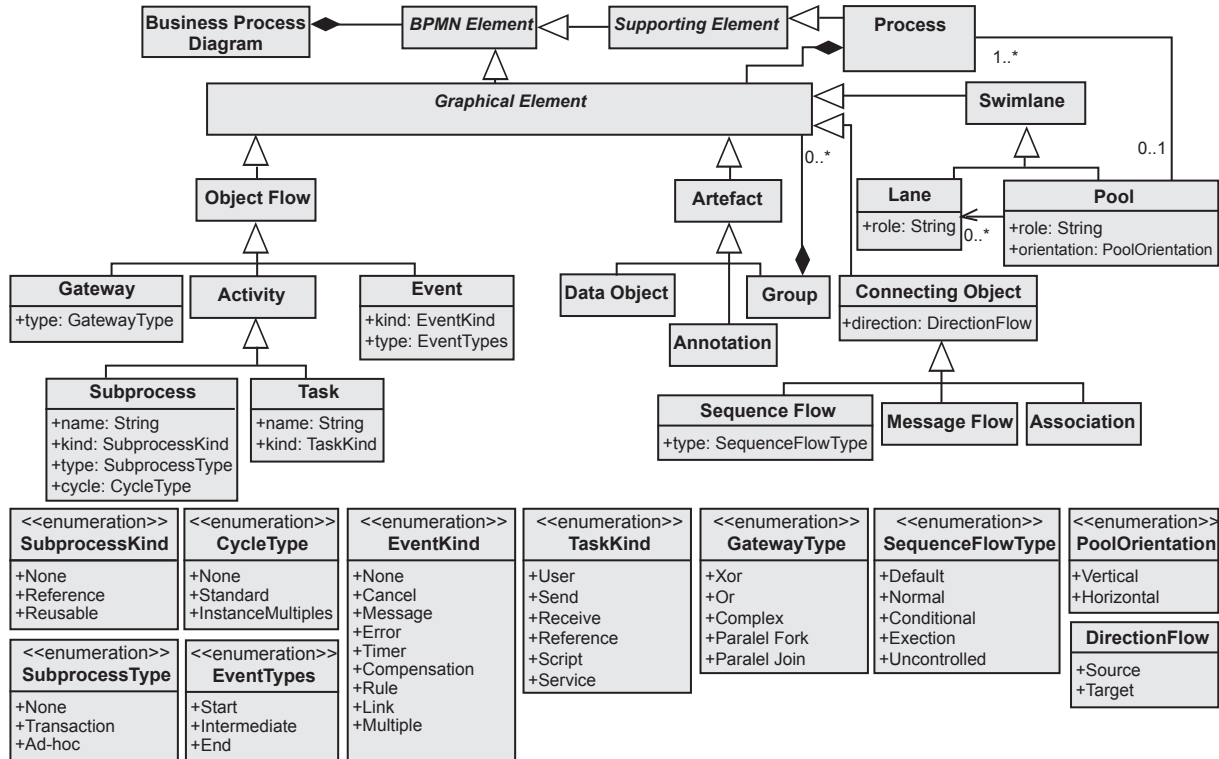
Hammer y Champy (1993) definen un *proceso* como una colección de actividades que toman una o varias clases de entradas y generan una salida de valor para el cliente. Desde el punto de vista del negocio, los procesos son ortogonales a las unidades gerenciales de una organización. Un *Proceso de Negocio* (PN) se considera un conjunto vinculado y natural de actividades basadas en habilidades y competencias de trabajo, que se inician desde los requisitos del cliente hasta la entrega total de productos o servicios (Chung, Gross y Yu, 1999). Un PN es más que una mera conexión de actividades, ya que incluye elementos tales como propósitos o intenciones, recursos utilizados, eventos, acciones, actividades, desencadenamiento de tareas y vínculos (Hepp y Roman, 2007). Las actividades de un PN son ejecutadas de acuerdo a reglas preestablecidas para satisfacer ciertas metas, las cuales son logradas a través de una o más tareas interrelacionadas, permitiendo obtener uno o más artefactos para una entidad concreta. White (2004) plantea que un *Modelo del Proceso de Negocio* (MPN) se define como una red de objetos gráficos, que incluyen actividades y controles de flujos que describen el funcionamien-

to de una organización. Los MPN son diseñados a través de lenguajes específicos, los cuales difieren en su naturaleza, característica y objetivos. Bajo esta premisa, el *Business Modeling Process Initiative Group* (BPMIG, 2002) estandarizó un conjunto de notaciones de modelado del negocio llamado Notación de Modelado de Procesos de Negocio o *Business Process Modeling Notation* (BPMN), unificando conceptos y puntos de vistas diferentes, aceptados recientemente por el *Object Management Group* (OMG, 2009). Consiste tanto de una notación gráfica así como también de la semántica subyacente.

El objetivo principal del lenguaje BPMN es ofrecer una notación entendible a todos los participantes del proceso de negocio (OMG, 2009). BPMN se basa en técnicas de *flowcharting* o diagrama de flujo y considera cuatro categorías: Objetos de Flujo, Objetos de Conexión, Carriles o *Swimlane*, los cuales se especializan en *Pools* y *Lanes*, Artefactos y dos elementos de modularización: Procesos y Diagramas del Proceso de Negocio (DPN) o *Business Process Diagram*, los cuales manejan la complejidad inherente a un proceso en particular.

En el **Gráfico 1**, presentamos un modelo conceptual de *BPMN* tomado de la OMG (2009), el cual se utiliza como marco de referencia para la elaboración y aplicación de los pasos y reglas para la correspondencia semántica que presentaremos en la sección 4, con la finalidad de derivar modelos de *GRL* desde diagramas de procesos de negocio de BPMN y viceversa.

**Gráfico 1**  
**Modelo conceptual de BPMN.**  
 OMG (2009)



## Lenguaje de requisitos orientado a metas (GRL)

En el contexto de la Ingeniería de Requisitos denominado orientación a metas (*goal-oriented approach*), el Lenguaje de Requisitos Orientado

a Metas o *Goal-oriented Requirements Language* (GRL) es un subconjunto de la Notación de Requisitos de Usuarios o *User Requirements Notation* (URN) (ITU-T, 2003; ITU-T, 2008), la cual admite la elicitación, el análisis, la especificación y la validación de requisitos, permitiendo descubrir y espe-

cificar requisitos de un sistema de software. *GRL* es un lenguaje orientado a metas y centrado en el razonamiento acerca de los requisitos, especialmente sobre los requisitos no funcionales y los relativos atributos de calidad que miden estos requisitos no funcionales. Proporciona construcciones para expresar los diferentes conceptos de la Ingeniería de Requisitos. *GRL* tiene sus raíces en dos enfoques de modelado orientado a metas, *i\** (Yu y Mylopoulos, 1993) y el *NFR Framework* (Chung, 1991).

El lenguaje *i\**, iniciado por Yu y Mylopoulos (1993), se centra en procesos de negocio, el cual admite la gestión sistemática e intencional de metas funcionales y no funcionales, partiendo del modelado de procesos de negocio, hasta llegar a los componentes arquitectónicos (Chung y Yu, 1998) (Chung, Gross y Yu, 1999). Está soportado por un lenguaje con una notación y semántica dinámica, fundamentada en la noción de intención de los actores, quienes actúan de manera autónoma, en el cumplimiento de metas (Yu, 1997). A diferencia de otras técnicas de modelado orientadas a metas, por ejemplo *KAOS*, (Van Lamsweerde, 2001), en las cuales las metas se descomponen durante el proceso de diseño en un modelo del proceso de negocio, en *i\** las metas se encuentran integradas en los actores (Yu, 1997). El enfoque *i\** tiene por objeto la identificación, representación, organización, análisis y justificación de metas tanto funcionales como no funcionales, bajo la noción de relaciones intencionales entre actores de un sistema (Yu, Strohmaier y Deng, 2006). También facilita el análisis del dominio y el modelado del ambiente representado por los actores y sus relaciones (Lapouchnian, 2005).

El *NFR (Non Functional Requirements Framework)* o Framework para Requisitos No Funcionales, iniciado por Chung (1991), es un enfoque orientado a metas el cual trata los requisitos no funcionales como *softgoals* u “objetivos blandos”, que se expresan en el *Softgoal Interdependence Graphic (SIG)* o Diagrama de Interdependencia de Softgoal. El SIG solo considera las metas no funcionales o *softgoals* en contraposición con los *hardgoals* u “objetivos duros”, que corresponden a metas funcionales. Los *softgoals* corresponden a objetivos no funcionales a partir de los cuales, a través de *operacionalizaciones (soluciones)*, se permiten derivar componentes arquitectónicos (Chung, Nixon y Yu, 1995; Chung y Yu, 1998). Una operacionalización corresponde a un mecanismo o componente arquitectónico que “satisface” o es una solución para el cumplimiento del *softgoal* (Chung, 1991; Chung, Nixon, y Yu, 1995).

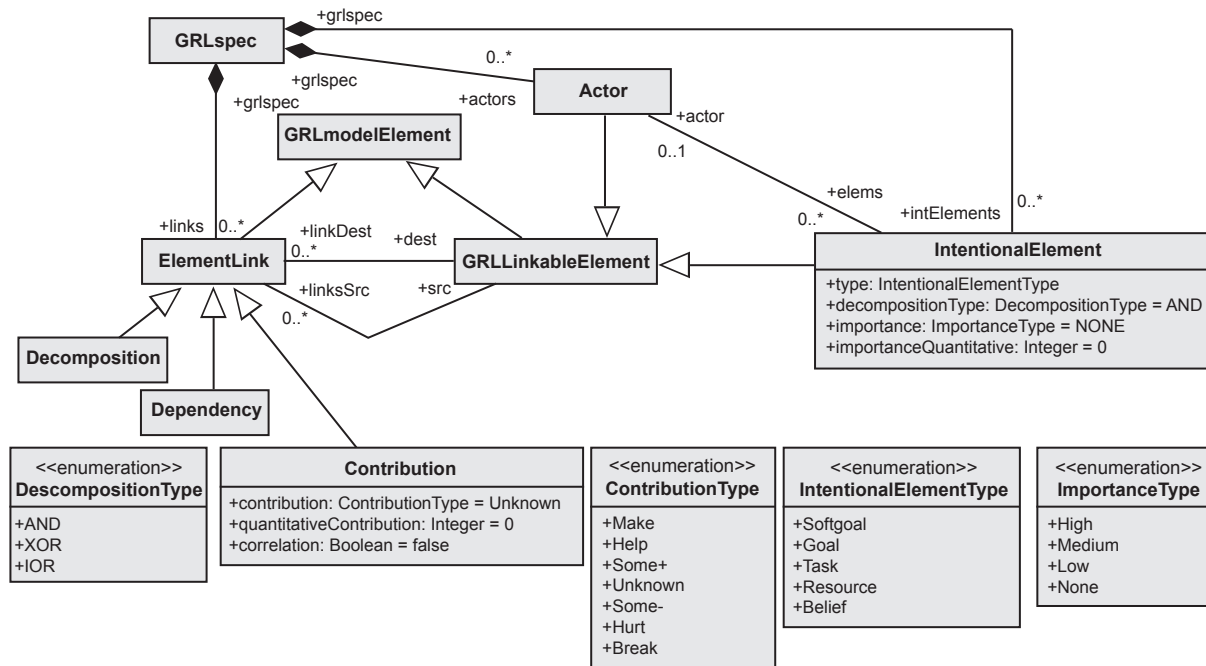
*GRL*, al integrar los citados lenguajes y enfoques orientados a metas, facilita la detección de aquellos requisitos que se originan en el contexto del negocio y deben ser trasladados al software y la resolución de conflictos a través de opciones alternativas de metas en competencia, facilitando una solución de diseño que satisfaga las metas u objetivos de calidad requeridos a nivel de negocio. En *GRL* existen tres categorías principales de conceptos, tales como: a) *actores*: son los titulares de las intenciones, entidades activas del sistema o de su entorno (p. e., los *stakeholders* u otros sistemas) que tienen intenciones y efectúan acciones para satisfacer metas funcionales o no y realizar tareas utilizando su *know-how* y los recursos que se encuentran disponibles. b) *elementos intencionales*:

en GRL están integrados por metas, *softgoals*, tareas, recursos y creencias, y c) *vínculos*: son usados para conectar elementos aislados del modelo de requisitos. Los diferentes tipos de vínculos representan las distintas relaciones estructurales e intencionales (incluyendo descomposiciones, contribuciones y dependencias).

Por otra parte, GRL soporta análisis de estrategias que ayudan a resolver de manera adecuada

los conflictos (*tradeoffs*) subyacentes entre las metas de los *stakeholders*. Una *estrategia* consiste en un conjunto de elementos intencionales que son de valor para la satisfacción inicial de la meta. Estos valores de satisfacción capturan situaciones contextuales o futuras, así como también permite elegir entre medios alternativos para alcanzar las referidas metas.

**Gráfico 2**  
**Modelo conceptual de GRL.**  
 ITU-T (2003)






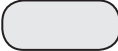
En el **Gráfico 2**, presentamos un modelo conceptual de *GRL* del ITU-T (2008) el cual se utiliza como marco de referencia para la correspondencia semántica presentada en la siguiente sección, en la cual se derivan diagramas de procesos de negocio de *BPMN* en modelos de *GRL*.


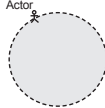

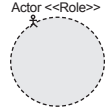

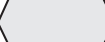

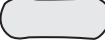
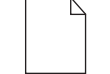





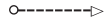

### Pasos y reglas para la correspondencia entre los lenguajes *BPMN* y *GRL*

En Cysneiros y Yu (2004) y Koliadis y otros (2007) se plantean algunas reglas de derivación para pasar de un modelo de *i\** a un modelo de *BPMN* en un contexto de reingeniería, es decir a partir de *I\** se reconstruyen los procesos de negocios. En nuestro trabajo, tomamos la regla “Identificación de Actores como *Pools*” de Cysneiros y Yu (2004) y la regla “Cualquier actor en un modelo

*i\** es un participante (*Pools* o *Lines*) en *BPMN* y viceversa” y el paso “Identificar los actores internos y externos en el diagrama de *i\** y viceversa” de Koliadis y otros (2007). Para incorporar a *BPMN* el concepto de meta no funcional, la cual puede ser exigida por un proceso con meta funcional, consideramos que las tareas en un proceso o subproceso de negocio, tienen asociadas metas funcionales y/o no funcionales en *GRL*. Identificamos la diferencia entre tareas funcionales y no funcionales en *BPMN* colocando una etiqueta <<MNF>> sobre el *vínculo* de asociación entre las tareas que poseen metas no funcionales (softgoals en *GRL*). Debe observarse que a un subproceso, que se compone de tareas, también podemos aplicar la asociación etiquetada si implica una meta no funcional. En el Cuadro 1 se presentan diez reglas para la correspondencia semántica entre *BPMN* y *GRL*.

**Cuadro 1**  
**Reglas de correspondencia semántica aplicables entre los lenguajes *GRL* y *BPMN***

| N°<br>Regla | Lenguaje <i>BPMN</i>    |   |   | Lenguaje <i>GRL</i>       |  |   | Regla para la Correspondencia  |
|-------------|-------------------------|---|---|---------------------------|--|---|--|
|             | Término                 | Definición  | Notación Gráfica  | Término                   | Definición   | Notación Gráfica  |  |
| 1           | Process o Proceso       | Actividad realizada dentro o a través de empresas u organizaciones (OMG, 2009). | Texto   |                           |  |   | Un <i>Proceso</i> de <i>BPMN</i> representa una <i>Meta Funcional</i> en <i>GRL</i> .  |
| 2           | Subprocess o Subproceso | Agregación de actividades que son incluidas dentro de un Proceso (OMG, 2009).   |  | Hardgoal o Meta Funcional | Condición o estado de un asunto que a los <i>stakeholders</i> les gustaría lograr (ITU-T, 2008). |  | Un <i>Subproceso</i> de <i>BPMN</i> representa una <i>Meta Funcional</i> que ha sido descompuesta en <i>GRL</i> en otras metas de nivel inferior |

|    |   |   |   |   |  |  |   |
|----|---|---|---|---|--|--|---|
| 3  | <i>Pool</i>                                 | Participantes (entidades o roles de negocio) (OMG, 2009).                                   |    | <i>Actor</i>  | Entidades activas que llevan a cabo ciertas acciones para lograr metas (ITU-T, 2008).  |    | Un <i>Pool</i> de BPMN representa un <i>Actor</i> en GRL.   |
| 4  | <i>Lane</i> o Carril                        | Organizan y categorizan actividades de un <i>Pool</i> (OMG, 2009).                          |    | <i>Role</i> o Papel   | Caracterización abstracta del comportamiento de un actor (Amyot y otros, 2009).  |    | Un <i>Carril</i> de BPMN es descrito como el <i>Papel</i> de un Actor concreto en GRL.                              |
| 5  | <i>Task</i> o Tarea                         | Es una actividad atómica (OMG, 2009).   |    | <i>Task</i> o Tarea   | Es una manera particular de hacer algo (ITU-T, 2008).  |    | Una <i>Tarea</i> de BPMN representa una <i>Tarea</i> particular en GRL.   |
| 6  | Labeled Association o Asociación Etiquetada | Muestra la asociación entre una tarea o subproceso funcional y una Meta No funcional, «MNF» |    | <i>Softgoal</i> o Meta No Funcional                                 | Es una condición o estado de un asunto del mundo que a los <i>stakeholders</i> les gustaría lograr (ITU-T, 2008).                  |    | Una Asociación Etiquetada de BPMN relaciona una tarea funcional o subproceso de BPMN con un <i>Softgoal</i> en GRL. |
| 7  | <i>Data Object</i> u Objeto de Datos        | Artefacto que suministra información a las actividades a ser realizadas (OMG, 2009).        |    | <i>Resource</i> o Recurso   | Entidad física o informacional que expresa necesidades (ITU-T, 2008).  |    | Un <i>Objeto de Datos</i> en BPMN representa un <i>Recurso</i> de GRL.  |
| 8  | <i>Gateway</i> o Compuerta                  | Controlan la interacción, convergencia o divergencia dentro de un proceso (OMG, 2009).      |   | <i>Task decomposition link</i> o Vínculo de descomposición de tarea | Tipo de vínculo que descompone una tarea en un <i>hardgoal</i> , <i>subtask</i> , <i>resource</i> o <i>softgoal</i> (ITU-T, 2008). |  | Una <i>Compuerta</i> en BPMN representa un <i>Vínculo de descomposición de tareas</i> (And/Xor/lor) de GRL.         |
| 9  | <i>Sequence Flow</i> o Flujo de Secuencia   | Muestra la secuencia de actividades que son realizadas en un proceso (OMG, 2009).           |  | <i>Means-Ends link</i> o Vínculos Medio-Fin                         | Relación binaria entre un fin y el medio para lograrlos. El “medio” representa una tarea y el “fin” una meta (ITU-T, 2008).        |  | Un <i>Flujo de Secuencia</i> en BPMN representa un <i>Vínculo Medio-Fin</i> en GRL.                                 |
| 10 | <i>Message Flow</i> o Flujo de Mensaje      | Muestra el flujo de mensaje entre dos entidades (OMG, 2009).                                |  | <i>Dependency link</i> o Vínculo de dependencia                     | Relación intencional entre dos actores (ITU-T, 2008).  |  | Un <i>Flujo de Mensaje</i> en BPMN representa un <i>Vínculo de Dependencia</i> entre dos actores en GRL.            |

A continuación se presentan cuatro pasos principales a seguir para lograr la referida correspondencia.

### **Pasos para la correspondencia entre BPMN y GRL**

**Paso 1.** En base al Diagrama del Proceso de Negocio (DPN) de BPMN, definir la Meta Funcional Medular (objetivo principal del proceso designada por los stakeholders) y posibles vínculos de dependencias del modelo de negocio en GRL Aplica la **Regla 1**.

**Paso 2.** En base a los Pools y los Lanes del DPN de BPMN, definir los Actores y Roles en GRL. Aplican las **Reglas 3 y 4**.

**Paso 3.** Identificar Pools, Lanes, Subprocesos, Tareas y asociaciones, Objetos de datos y Compuertas en el DPN de BPMN y hacerlas corresponder con Metas Funcionales, Metas No Funcionales y Tareas en el modelo de negocio de GRL:

**Paso 3.1.** Detectar Metas No Funcionales que estén asociadas a aspectos no funcionales del negocio (como por ejemplo algunas reglas del negocio, restricciones de plataforma) en BPMN y asociarlas con las Metas No Funcionales o softgoals de GRL. Aplica la **Regla 6**. Debe observarse que en BPMN no existe una notación explícita para expresar posibles restricciones o metas no funcionales relacionadas con un proceso o subproceso de negocio; por lo tanto consideramos que las tareas pueden ser utilizadas con este fin: las tareas no

funcionales, se distinguen de las funcionales por el vínculo de asociación etiquetado por <<MNF>>. A un subproceso también puede aplicarse la asociación etiquetada. En particular una regla del negocio puede ser un *softgoal* potencial, por ejemplo exigir control de acceso para cierto tipo de usuarios, es un *softgoal* relacionado con tener seguridad para el acceso de usuarios.

• **Paso 3.2.** Asociar Tareas, Objetos de datos y Compuertas expresados en BPMN con Tareas, Recursos y Vínculos de Descomposición de Tareas en GRL. Aplican las **Reglas 5, 7 y 8**.

• **Paso 3.3.** Los subprocesos expresados de BPMN se definen como una descomposición de una Meta Funcional en GRL y los elementos subyacentes son representados de acuerdo a las reglas y pasos que tengan lugar. Aplica la **Regla 2** y nuevamente la **Regla 6**, de ser el caso.

**Paso 4.** En base a los diferentes elementos que se encuentran presentes en los Pools y los Lanes del DPN de BPMN, establecer vínculos en GRL:

• **Paso 4.1.** Definir Vínculos de Dependencia entre los diferentes elementos de GRL. Aplica la **Regla 10**.

• **Paso 4.2.** Generar vínculos medio-fin en GRL a partir de flujos de secuencias explícitos en BPMN. Aplica la **Regla 9** y la experiencia del Ingeniero de Requisitos.

• **Paso 4.3.** Identificar los Vínculos de Correlación entre Metas No Funcionales en GRL.

• **Paso 4.4.** Establecer las Contribuciones entre las Metas Funcionales o No Funcionales en GRL.

## Trabajos relacionados

Cysneiro y Yu (2004) muestran cómo  $i^*$  puede ser usado como *front-end* a las tecnologías de modelado del proceso de negocio para incorporar autonomía en los actores y disminuir la brecha entre estos y los lenguajes de modelado del proceso de negocio, tales como *BPMN*: una vez establecido el proceso en  $i^*$ , éste se deriva a *BPMN*. Koliadis y otros (2007) plantean que el manejo de cambios en el ciclo de vida de un proceso de negocio puede ser soportado efectivamente combinando diferentes notaciones. El trabajo intenta representar los cambios realizados de un modelo  $i^*$  a un modelo de *BPMN*. En Koliadis y Ghose (2007) se propone un enfoque metodológico para vincular modelos basados en *BPMN* a metas de alto nivel usando *KAOS (Knowledge Acquisition in autOmated Specification)* (Van Lamsweerde, 2001), pero no consideran a  $i^*$ . En vista de que los requisitos no funcionales no son explícitamente tratados en *BPMN*, Pavlosky y Zou (2008) proponen una clasificación de requisitos no funcionales asociados al proceso de negocio, en *operacionales* (rendimiento, confiabilidad, seguridad, tiempo de respuesta, calidad del servicio y disponibilidad, sin especificar las definiciones de estos requisitos) y *no operacionales* que incluyen consideraciones relativas a la tecnología; proponen dos conceptos, la *condición de operación* y el *caso de control*, respectivamente, para capturar estos requisitos de alto nivel durante el modelado del proceso del negocio. Di Francescomarino y Tonela (2008) proponen el uso de *Semantic Annotations* o Anotaciones Semánticas como mecanismos para tratar los requisitos no funcionales asociados a las actividades donde

tales restricciones son aplicadas. Nuestro aporte respecto a los enfoques citados consiste en haber utilizado GRL en lugar de  $I^*$ , lenguaje más general que engloba a  $I^*$ , y proponer el uso de tareas y las asociaciones etiquetadas como mecanismos para tratar los aspectos no funcionales. En este sentido, ampliamos los referidos enfoques para el establecimiento de los pasos y reglas necesarias para establecer la correspondencia semántica entre los lenguajes *BPMN* y *GRL*, de manera de reducir la brecha existente entre las metas organizacionales y sus sistemas de software, destacando aspectos no funcionales en las etapas tempranas del ciclo de vida del desarrollo de software. Esta correspondencia puede ser utilizada en un contexto de reingeniería e ingeniería de modelos, particularmente para la derivación de un modelo arquitectónico del sistema a partir de un modelo de negocio.

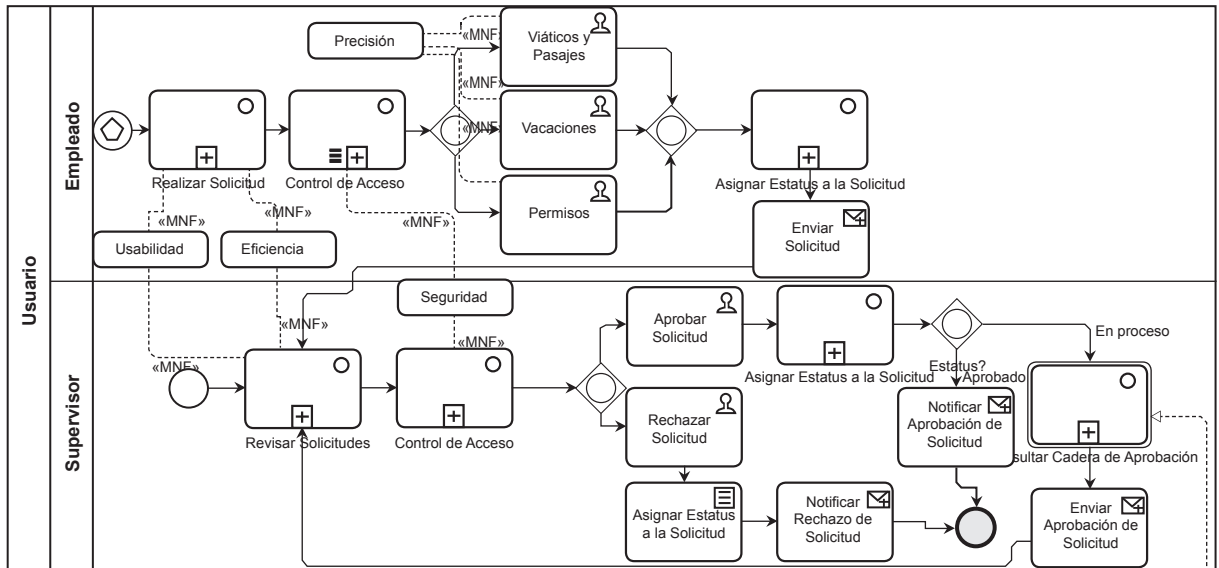
## Caso de estudio

Se considera un DPN del proceso de Gestión de Trámites de Solicitudes (GTS) de la Dirección del Talento Humano, que involucra a las Unidades Medulares del Centro Nacional de Tecnologías de Información (CNTI). El proceso GTS permite el establecimiento de flujos de trabajo o workflows, según la estructura organizativa en sus niveles gerenciales con el propósito de automatizar los procesos de solicitudes de: Viáticos (asignación financiera para manutención) y Pasajes, Vacaciones, Permisos, Préstamos (implica el requisito no funcional *precisión*), entre otras. La meta funcional medular, de alto nivel organizacional, está centrada en Gestionar Trámites de Solicitudes por medio de redes seguras (implica el

requisito no funcional *seguridad*) con el propósito de garantizar la satisfacción del usuario final (implica el requisito no funcional *usabilidad*) y disminuir los tiempos de respuestas de las solicitudes a lo sumo tres días (implica el requisito no funcional *eficiencia*). Se aplicarán las reglas y pasos descritos en la sección anterior para derivar el modelo GRL de un sistema automatizado para GTS, a partir del diagrama del proceso de negocio de GTS expresado en BPMN.

En la **Figura 1**, se observa el DPN GTS expresado en *BPMN* mediante la herramienta *BizAgi Process Modeler* (BizAgi Ltd, 2008). A continuación se aplican las reglas de correspondencia semántica del cuadro 1 al citado caso de estudio, para derivar un Modelo en *GRL* explicitado en la Figura 2, utilizando como herramienta *jUCMNav* (Mussbacher, 2010), donde utilizamos la notación *BPMN* de la *OMG* (2009) y la notación *GRL* del *ITU-T* (2008):

**Figura 1**  
**Modelo de negocio GTS expresado en BPMN**



## Aplicación de las reglas de correspondencia semántica al proceso de negocio de GTS

**Paso 1.** En base al Diagrama del Proceso de Negocio (DPN) de BPMN, definir la Meta Funcional Medular (objetivo principal del proceso designada por los stakeholders) y posibles vínculos de dependencias del modelo de negocio en GRL. Basándonos en la **Regla 1**, definimos la meta principal del proceso de negocio GTS, denominado Gestionar Trámites de Solicitudes y el nombre del modelo de proceso de negocio en GRL, denominado Modelo de Negocio de Gestión de Trámites de Solicitudes.

**Paso 2.** En base a los Pools y los Lanes del DPN de BPMN, definir los Actores y Roles en GRL. Los actores derivados de la **Regla 3**, son: Usuario, Ingeniero de Sistemas, Gestor de Solicitudes y Administrador. De la aplicación de la **Regla 4**, definimos dos roles en el actor Usuario ya que este juega los roles de Empleado y de Supervisor (ver **Figura 2**).

**Paso 3.** Identificar Pools, Lanes, Subprocesos, Tareas y asociaciones, Objetos de datos y Compuertas en el DPN de BPMN y hacerlas corresponder con Metas Funcionales, Metas No Funcionales y Tareas en el Modelo de Negocio de GRL:

**Paso 3.1.** Detectar tareas funcionales y que estén asociadas a tareas no funcionales en BPMN y colocar la asociación etiquetada <<MNF>> y asociarlas luego con las Metas No Funcionales o softgoals de GRL. A continuación se describen los elementos que subyacen a las Me-

tas No Funcionales del proceso de GTS, derivados de la aplicación de la **Regla 6**:

*Lane Usuario-empleado:* tareas funcionales: *Viático y pasaje, Vacaciones y Permisos*, que tienen asociada la tarea no funcional *precisión; Enviar solicitud*

*Lane Usuario-supervisor:* tarea funcional: *Aprobar, Rechazar, Asignar estatus;*

Los Lanes Usuario-Administrador, Usuario-Gestor de Solicitudes, y Usuario-Ingeniero de Sistemas no se muestran aquí para no recargar la presentación, sin embargo se repite el proceso descrito para todos estos elementos.

**Paso 3.2.** Asociar Tareas, Objetos de datos y Compuertas expresados en BPMN con Tareas, Recursos y Vínculos de Descomposición de Tareas en GRL. A continuación se describen los elementos derivados de la aplicación de las **Reglas 5, 7 y 8**:

*Usuario-empleado: Tareas:* Viáticos y Pasajes; Vacaciones; Permisos y Enviar solicitud. *Vínculos de descomposición de tareas:* Tipo de solicitud y Estatus.

*Usuario-supervisor: Tareas:* Aprobar Solicitud, Rechazar Solicitud, Notificar Aprobación de Solicitud, Enviar Aprobación de Solicitud, Asignar Estatus a la Solicitud, Notificar Rechazo de Solicitud. *Vínculos de descomposición de tareas:* Aprobación y Estatus.

**Paso 3.3.** Los subprocesos expresados de BPMN se definen como Metas Funcionales en GRL. Si involucran tareas no funcionales, se aplica nuevamente la **Regla 6**.



*Usuario-supervisor: Metas Funcionales:* Revisar Solicitudes, Control de Acceso, Asignar Estatus a la Solicitud y Consultar Cadena de Aprobación.

Los subprocesos presentados a continuación no son desarrollados en este trabajo para no recargar la presentación, pero siguen el mismo proceso descrito anteriormente.

*Gestor de Solicitudes: Metas Funcionales:* Monitorear Solicitudes.

*Administrador: Metas Funcionales:* Gestionar Usuarios, Control de Acceso y Asignar Permisos al Usuario.

*Ingeniero de Sistemas: Metas Funcionales:* Gestionar *Workflow*, Control de Acceso, Configurar *Workflow*.

**Paso 4.** *En base a los diferentes elementos que se encuentran presentes en los Pools y los Lanes del DPN de BPMN, establecer vínculos en GRL:*

**Paso 4.1.** *Definir Vínculos de Dependencia entre los diferentes elementos de GRL.* Se definieron los Vínculos de Dependencia implícitos entre las tareas, recursos, metas funcionales y no funcionales en base a la **Regla 10** (ver **Figura 2**).

**Paso. 4.2.** *Generar vínculos medio-fin en GRL a partir de flujos de secuencias explicitados en BPMN.* En el actor Administrador se definió un Vínculo Medio-Fin que tiene como medio Crear, Modificar y Eliminar Cuenta y como fin Gestionar Usuario, en base a la **Regla 9**.

**Paso. 4.3.** *Identificar los Vínculos de Correlación entre Metas No Funcionales en GRL.* Se definieron los vínculos de correlación entre las metas no funcionales Usabilidad, Eficiencia, Precisión y Seguridad (ver **Figura 2**).

**Paso. 4.4.** *Establecer las Contribuciones entre las Metas Funcionales o No Funcionales en GRL.* Se definieron las contribuciones entre las metas funcionales y no funcionales especificadas en los pasos anteriores. En la Figura 2 se muestran las metas satisfechas (✓), poco satisfechas (✓.), en conflictos (✗), poco denegadas (✗.), denegadas (✗) y desconocidas (?).

## Conclusiones

El lenguaje *BPMN*, centrado en eventos y dirigido a aspectos esencialmente funcionales, facilita la comprensión del dominio del problema para plantear soluciones funcionales desde una perspectiva clásica de la Ingeniería de Negocio. Es muy utilizado en la práctica en la industria para especificar procesos de negocio. Sin embargo, carece de una notación explícita para representar aspectos no funcionales, los cuales son imprescindibles para derivar el modelo arquitectural inicial del sistema de software que corresponde al proceso de negocio que se quiere automatizar. Así mismo, es ahora de común aceptación que los aspectos no funcionales deben ser considerados temprano en el proceso de desarrollo, para evitar dificultades posteriores de mantenimiento. Por otra parte, el lenguaje *GRL* de naturaleza lógica, dirigido por objetivos y orientado al cumplimiento de metas por actores, explícita las dependencias



entre actores y el razonamiento asociado y destaca también aspectos no funcionales, los cuales no son contemplados en BPMN. Para poder expresar aspectos no funcionales en etapas tempranas del ciclo del software, en particular desde la perspectiva del negocio que requiere el sistema, se presenta una correspondencia semántica entre los lenguajes *BPMN* y *GRL*, la cual puede ser utilizada para cubrir la deficiencia de BPMN en cuanto al tratamiento de metas no funcionales, ayudando a reducir la brecha entre las metas organizacionales y sus sistemas de software. Utilizamos la noción de tareas no funcionales (asociadas a metas no funcionales) como mecanismos para tratar los *softgoals* de *GRL* en *BPMN*. Diferenciamos entre tareas funcionales y no funcionales, lo cual está supeditado al tipo de vínculo utilizado en la tarea: para representar las tareas no funcionales utilizamos los vínculos de asociación. Así mismo, esbozamos que las reglas de correspondencia aquí propuestas pueden ser utilizadas en un contexto de reingeniería de procesos tal como en el caso de estudio presentado en este trabajo y como marco de referencia para la generación de transformaciones de un diagrama del proceso de negocio de *BPMN* a un modelo de *GRL* y viceversa. En referencia al caso de estudio, utilizamos un proceso de negocio del mundo real correspondiente a la Gestión de Trámites de Solicitudes del CNTI. En trabajos futuros proponemos extender los pasos y reglas para la correspondencia semántica aquí propuesta para refinar y tratar metas no funcionales expresadas en *GRL* como incumbencias transversales candidatas a aspectos para llegar a un modelo de arquitectura inicial, considerando un enfoque orientado a aspectos.

## Agradecimiento

Queremos agradecer al PhD. Daniel Amyot, miembro activo del Sector de Estandarización de las Telecomunicaciones de la organización *International Telecommunication Union* (ITU) por habernos suministrado copia de los estándares z. 150 (2003) y z. 151 (2008) de esa prestigiosa institución. Así mismo agradecemos a los earbitros por la cuidadosa lectura y el esfuerzo hecho para mejorar la calidad de este trabajo.

## Bibliografía

- Amyot, D., Horkoff, J., Gross, D. y Mussbacher, G. (2009). *A Lightweight GRL Profile for i\* Modeling*. Heidelberg, Berlin: Springer.
- BizAgi Ltd. (2008). *BPM BizAgi*. Disponible en: <http://www.bizagi.com/>. [Recuperado: 24 de Marzo de 2010].
- BPMIG. (2002). *Business Process Modeling Language*. GLiNTECH, pp.1-45.
- Burlton, R. (2001). *Effective Business Change through Process Management: Strategies and Architectures for Integrated Change*. Process Renewal Group.
- Chung, L. (1991). *Representation and Utilization of Non\_Functional Requirements for Information System Design*. CAiSE 91, 3rd Int. Conf. Advanced Information Systems Eng. (pp. 5-30). Norway, Berlin: Springer-Verlag.
- Chung, L. y Yu, E. (1998). *Achieving System-Wide Architectural Qualities*. Proc. of OMG-DARPA: MCC Workshop on Compositional Software Architectures. Monterey, California, USA: CiteSeerX.

- Chung, L., Gross, D., y Yu, E. (1999). *Architectural design to meet stakeholder requirements*. In Donohue Software architecture (pp. 545-564). Texas, USA: Kluwer Academic Publishers.
- Chung, L., Nixon, B., y Yu, E. (1995). *Using Non-Functional Requirements to Systematically Select Among Alternatives in Architectural Design*. Proc. of 1st Int. Workshop on architectures for Software System (pp. 31-32). Seattle, Washington: CiteSeerX.
- Crusson, T. (2006). Business Process Management Essentials. *Glintech*.
- Cysneiros, L., y Yu, E. (2004). *Addressing Agent Autonomy in Business Process Management - with Case Studies on the Patient Discharge Process*. Proc. of the Information Resources Management Association Conference. New Orleans, USA: ACM.
- Di Francescomarino, C., y Tonella, P. (2008). *Supporting Documentation and Evolution of Crosscutting Concerns in Business Processes*. Proc. ICSOC PhD Symposium 2008 collocated with 6th International Conference on Service Oriented Computing (ICSOC). Sydney, Australia: CiteSeerX.
- Hammer, M., y Champy, J. (1993). *Reengineering the Corporation*. New York, US: Harper Collins.
- Hepp, M., y Roman, D. (2007). *An Ontology Framework for Semantic Business Process Management*. Proc. of the 8th international conference Wirtschaftsinformatik (pp. 1-18). Karlsruhe: Springer.
- ITU-T. (2008). *User Requirements Notation (URN)-Language Definition: Recommendation ITU-T Z.151*. ITU, pp. 1-206.
- ITU-T. (2003). *User Requirements Notation (URN)-Language requirements and framework: Recommendation ITU-T Z.150*. ITU, pp. 1-29.
- Koliadis, G., y Ghose, A. (2007). *Relating Business Process Models to Goal-Oriented Requirements Models in KAOS*. Proc. of the Pacific-Rim Knowledge Acquisition Workshop. Australia: The Berkeley Electronic Press.
- Koliadis, G., Vranesevic, A., Bhuiyan, M., Krishna, A., y Ghose, A. (2007). *A combined approach for supporting the business process model lifecycle*. Proc. of Asia-Pacific Conference on IS (pp. 1-13). Wollongong: Computing Science.
- Lapouchnian, A. (2005). *Goal-Oriented Requirements Engineering: An Overview of the Current Research*. Toronto: University of Toronto.
- Mussbacher, G. (2010). *jUCMNav*. Disponible en: <http://jucmnav.softwareengineering.ca/ucm/bin/view/ProjetSEG/WebHome> . [Recuperado: 15 de Octubre de 2010].
- OMG. (2009). *Business Process Modeling Notation Specification*, Ver. 1.2. OMG , pp. 1-294.
- Pavlovski, C. y Zou, J. (2008). Non-Functional Requirements in Business Process Modeling. In Proc. Fifth Asia-Pacific Conference on Conceptual Modelling (APCCM 2008), Wollongong, NSW, Australia. CRPIT, 79. Hinze, A. and Kirchberg, M., Eds. ACS. 103-112.
- Van Lamsweerde, A. (2001). *Goal-oriented requirements engineering: A guided tour*. Proc. of RE'01 - International Joint Conference on Requirements Engineering (pp. 249-263). Toronto: IEEE.
- White, S. (2004). *Business Process Modeling Notation (BPMN), Version 1.0*. Business Process Management Initiative (BPMI.org). 1.0 edn.

- Yu, E. (1997). *Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering*. Proc. of the 3rd IEEE Int. Symp. on Requirements Engineering (pp. 226-235). IEEE Computer Society Washington, DC, USA: IEEE.
- Yu, E. (1997). *Why Agent-Oriented Requirements Engineering*. Proc. 3rd Int. Workshop on Requirements Engineering: Foundations of Software Quality REFSQ'97, (pp. 171-183). Barcelona, Catalonia, Spain.
- Yu, E., y Mylopoulos, J. (1993). *An Actor Dependency Model of Organization Work-With Application to Business Process Reengineering*. Proc. Conf. on Organizational Computing Systems (COOCS '93) (pp. 258-268). California: IEEE.
- Yu, E., Strohmaier, M., y Deng, X. (2006). *Exploring Intentional Modeling and Analysis for Enterprise Architecture*. Proc. of the EDOC 2006 Conference Workshop on Trends in EnterpriseArchitecture Research. Hong Kong: IEEE Computer Society.