

UNIVERSIDAD CENTRAL “MARTA ABREU” DE LAS VILLAS
FACULTAD MATEMÁTICA, FÍSICA Y COMPUTACIÓN
INGENIERÍA INFORMÁTICA



Trabajo de Diploma

*Algoritmo para la transformación al formato ORM2 de
un conjunto de reglas de negocio expresadas sobre SBVR
para la detección de inconsistencias*

Autor: Lázaro González Isidor

Tutores: Dra. María Elena Martínez Busto

Lic. Víctor Marrero Shimko

Curso Académico: 2013-2014

Dictamen

Hago constar que el presente trabajo fue realizado en la Universidad Central “Marta Abreu” de Las Villas como parte de la culminación de los estudios de la especialidad de Ingeniería Informática, autorizando a que el mismo sea utilizado por la institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos ni publicado sin la autorización de la Universidad.

Firma del autor

Los abajo firmantes, certificamos que el presente trabajo ha sido realizado según acuerdos de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

Firma del tutor

Firma del jefe del Seminario

Pensamiento

Pues la mente humana suele andar a menudo tan torpe y desconcertada en esta pista de los inventos, que primero desconfía y poco después se desdeña a sí misma; y así al principio le parece increíble que pueda descubrirse semejante cosa, más luego de descubierta le parece también increíble que una cosa así, haya podido escapar a los hombres por tanto tiempo.

Francis Bacon

Dedicatoria

A mis padres por guiarme en todo el camino

A mis abuelos por su eterno cariño

A mi esposa Eviani por ser el precioso regalo de Dios

Agradecimientos

A Dios por sobre todas las cosas

A mis tutores por todo su apoyo, en especial a Víctor

A mis familiares por toda su ayuda

A mi tío Félix que siempre lo llevo en el corazón

A aquellos que hicieron posible la realización de este trabajo

Resumen

El enfoque de reglas de negocio está dirigido al desarrollo de aplicaciones más flexibles y adaptables ante los cambios que se producen en las políticas del negocio. Repetidas veces, estos cambios percibidos sobre las reglas que rigen el negocio, se producen sin llevar a cabo un análisis consciente de las afectaciones que pueden acarrear. SBVR (*Semantics of Business Vocabulary and Business Rules*) es un estándar que sirve de guía para la representación de reglas y vocabularios de negocio. En el presente trabajo se propone un algoritmo de transformación de un conjunto de reglas de negocio expresadas en SBVR al formato ORM2 (Object Relational Mapping 2) para la detección de inconsistencias por parte de los expertos del negocio. Dentro de las perspectivas se encuentra el enriquecimiento de una herramienta ya confeccionada para realizar dichas operaciones con el objetivo de hacer más consistente el repositorio. El software existente permite a los usuarios del negocio la manipulación de un conjunto de reglas previamente creado. Finalmente, se obtiene un algoritmo que transforma reglas de negocio expresadas en SBVR hacia el formato ORM2 siguiendo una serie de pasos que quedan plasmados para su implementación y su integración con la herramienta de Edición de reglas y vocabulario (UcRules).

Abstract

The focus of business rules is directed to the development of more flexible and more adaptive applications in the face of the changes that take place in the politicians of the business. Repeated times, these changes perceived on the rules that govern the business, take place without carrying out an analysis aware of the affectations that can carry. SBVR (Semantics of Business Vocabulary and Business Rules) is a standard that serves as guide for the representation of rules and business vocabularies. Presently work intends an algorithm of transformation of a group of business rules expressed in SBVR to the format ORM2 (Object Relational Mapping 2) for the detection of inconsistencies on the part of the experts of the business. Inside the perspectives he/she is already the enrichment of a tool made to carry out this operations with the objective of making more consistent the repository. The existent software allows the users of the business the manipulation of a previously created group of rules. Finally, an algorithm is obtained that transforms business rules expressed in SBVR toward the format ORM2 following a series of steps that you/they are captured for its implementation and its integration with the tool of Edition of rules and vocabulary (UcRules).

Contenido

Introducción	1
CAPÍTULO 1. MODELADO DE REGLAS DE NEGOCIO	6
1.1. Reglas de Negocio.....	6
1.2. Notación Lineal ORM2 como soporte de Validación para SBVR.....	7
1.3. Tipos de Restricciones y su especificación en la notación lineal ORM2.....	8
1.4. Estándar para la representación de reglas y vocabularios de negocio.....	17
1.4.1. Fundamentos Básicos de SBVR	18
1.4.2. Formalización Semántica de SBVR.....	20
1.4.3. Problemas en la formalización de SBVR.....	20
1.4.4. Operadores lógicos usados en SBVR	21
1.5. Conclusiones parciales del capítulo	24
CAPÍTULO 2. FORMALIZACIÓN DE UN ALGORITMO PARA LA TRANSFORMACIÓN DE REGLAS DE NEGOCIO EXPRESADAS EN SBVR A ORM2 ...	26
2.1. Transformación de Reglas de Negocio	26
2.2. Validación y verificación de reglas de negocio	31
2.3. Algoritmo formal.....	32
2.4. Conclusiones parciales del capítulo	35
CAPÍTULO 3. EJEMPLO DE APLICACIÓN DEL ALGORITMO PARA EL CASO DE ESTUDIO: CONTROL DE PACIENTES EN EL ÁREA DE NEFROLOGÍA.	37
3.1. Caso de estudio: control de pacientes en el área de nefrología	37
3.1.1. Control de los pacientes en el área de nefrología	38
3.1.2. Caracterización del caso de estudio	41
3.2. Aplicación del algoritmo a una muestra del caso de estudio.	42

Conclusiones	48
Recomendaciones	49
Referencias bibliográficas.....	50

Índice de Tablas

Tabla 1-1 Notación lineal ORM2	12
Tabla 1-2 Cuantificadores de la lógica de predicados	22
Tabla 1-3 Operadores lógicos proposicionales	23
Tabla 1-4 Operadores Modales	24

Introducción

Los sistemas automatizados que modelan los distintos procesos que ocurren en las empresas se han convertido en un tema de interés para organizaciones que buscan soluciones de almacenamiento, distribución y análisis del conocimiento de los procesos del negocio. Uno de los enfoques más comunes para describir el negocio y la información usada por el mismo es basado en reglas.

Este enfoque consiste en identificar y formular reglas que definen la estructura y el control de las operaciones de una empresa. Por ello la principal expectativa en la implementación de soluciones automatizadas del enfoque basado en reglas es la habilidad de determinar automáticamente la consistencia de las reglas en un modelo de negocio.

Las reglas de negocio son definiciones explícitas que regulan cómo opera un determinado negocio y cómo el mismo es estructurado (Bajec, Krisper, & Rupnik, 2000; Hay, Kolber, & Anderson, 1997; Youdeowei, 1997) en este contexto, se entiende por negocio cualquier tipo de servicio que ofrezca determinada institución. Las reglas de negocio son imprescindibles tanto para el funcionamiento de la empresa o institución correspondiente, así como para el sistema de información que soporta sus procesos.

La forma de captar las reglas en cualquier entidad u organización debe realizarse en un lenguaje natural restringido, de la forma en que la comunidad de usuarios expresan sus regulaciones. Esto no es suficiente, debido a que una vez capturadas pueden no reflejar ideas precisas o incluso pueden resultar contradictorias entre sí, por tanto se requiere hacer un tratamiento adicional al repositorio de reglas y realizar una validación sobre este.

El propósito final desde el enfoque de reglas de negocio es que el usuario del negocio tenga el control directo sobre la definición de sus reglas. Actualmente esta no es una opción totalmente lograda pero es definitivamente la dirección que se debe seguir (Morgan, 2002).

Pueden usarse varias vías para expresar las reglas de negocio en un formato consistente. Generalmente mediante un vocabulario estándar conformado por palabras y definiciones establecidas por la comunidad de usuarios de un negocio particular (Ronald G. Ross, 2003).

En la literatura revisada se explica, desde una perspectiva teórica y de forma relativamente sencilla, la creación de reglas de negocio comprensibles para los usuarios del negocio a partir de un vocabulario base (Morgan, 2002; Weiden, Hermans, Schreiber, & van der Zee, 2004). Al

analizar el problema desde una perspectiva práctica, la creación de dicho vocabulario se torna compleja, así como definir una vía factible para que una persona, no entrenada en sistemas informáticos, pueda comprender sus especificidades. Actualmente no se encuentran criterios de solución unificada que aborde el problema salvando ambos obstáculos.

La Semántica para el Vocabulario y las Reglas del Negocio (SBVR) (OMG, 2008) está formalmente definida como una taxonomía que describe operaciones y reglas de negocio elementales. Es una especificación propuesta inicialmente por el Grupo de Trabajo para la Estandarización en el Ámbito de las Reglas de Negocio (BRG) y tomada por el Grupo de Estandarización en su meta modelo expresado en Lenguaje Unificado de Modelado (UML), integrado dentro de lo que se denomina Arquitectura Dirigida por Modelos (MDA) (OMG, 2008). Detalla cómo las reglas y vocabulario de negocio se pueden capturar y especificar en términos no ambiguos en un lenguaje lógico formal con expresiones formuladas en un lenguaje natural restringido.

El vocabulario de SBVR permite que se especifiquen formalmente representaciones de conceptos, de definiciones, de casos, y de reglas de cualquier dominio del conocimiento en lenguaje natural, incluyendo formas tabulares. Estas características hacen SBVR bien adaptado para describir dominios del negocio.

SBVR permite el desarrollo multilingüe, puesto que se basa en la separación entre los símbolos y su significado. Permite además la fabricación de reglas de negocio accesibles a las herramientas del software, incluyendo las herramientas que apoyan a expertos del negocio en crear, encontrar, y validar reglas.

Provee además una capacidad jerárquica de categorización para organizar conceptos desde lo general hasta lo más específico, lo cual se le conoce como taxonomías. En SBVR las reglas son definidas aplicando necesidades u obligaciones a los tipos de hechos.

La validación de la consistencia y los conflictos es una de las tareas más delicadas y consumidoras de tiempo en las investigaciones de reglas de negocio, tanto como la semántica de las reglas especificada en el lenguaje natural, que resultan esenciales para comprender y desarrollar los sistemas de información (Ronald G. Ross, 2000).

En el Grupo de Bases de Datos del Centro de Estudios Informáticos se han desarrollado trabajos de investigación con el objetivo de implementar herramientas para la gestión y validación de reglas de

negocio. En el trabajo de diploma con título “Administración de reglas de negocio desde la perspectiva del negocio basada en SBVR” (Guedes, 2013), se desarrolla una herramienta para la edición de vocabularios y reglas de negocios. En la tesis de maestría con título “Validación de la consistencia de un conjunto de reglas de negocio” (Núñez, 2012), se describen algunas pautas para la obtención de un repositorio de reglas consistente mediante la aplicación de determinados mecanismos de validación. En el trabajo de diploma “Tablas de decisión para el análisis de consistencia y chequeos de integridad de un conjunto de reglas de negocio expresadas en SBVR” (Marrero, 2013), se obtiene una herramienta para la validación de un conjunto de reglas de negocio mediante la utilización de tablas de decisión. Aún se presentan brechas en las técnicas de validación implementadas para la obtención de un repositorio de reglas totalmente consistente. En este sentido se presenta la necesidad de enriquecer la herramienta de Edición y Vocabulario de reglas de negocio(UcRules)(M. E. Martínez Busto, Caballero Martínez, A., Guedes Coello, D., Nuñez, C. A. & González González, L, 2012) con nuevos métodos de validación.

Por lo cual se define como **objetivo general de investigación** formalizar un algoritmo para la transformación de reglas de negocio expresadas sobre SBVR al formato ORM2 para la detección de inconsistencias por parte de expertos del negocio.

Este queda desglosado en los siguientes **objetivos específicos**:

1. Formalizar el algoritmo para implementar las transformaciones de las reglas de negocio escritas sobre SBVR al formato ORM2.
2. Aplicar el algoritmo de transformación de reglas de negocio escritas sobre SBVR al formato ORM2 mediante una muestra del caso de estudio: Control de pacientes en el área de nefrología.
3. Conformar el caso de estudio transformando un conjunto de reglas de negocio escritas sobre SBVR acorde a las descripciones propuestas en investigaciones precedentes.

Las **preguntas de investigación** planteadas son:

¿Cómo queda formalizado el algoritmo para implementar las transformaciones de las reglas de negocio escritas sobre SBVR al formato ORM2?

¿Qué inconsistencias e irregularidades se encontraron en el conjunto de reglas de negocio del caso de estudio durante el proceso de validación de las reglas transformadas a ORM2?

La tesis está estructurada en 3 capítulos. En el Capítulo 1 se aborda la modelación de las reglas de negocio desde la perspectiva de SBVR, así como la notación lineal ORM2, definiendo aspectos fundamentales de SBVR, las reglas de negocio, los tipos de restricciones de ORM2 y elementos principales. En el Capítulo 2 se presenta una serie de transformaciones para un conjunto de reglas de negocio escritas sobre SBVR acorde a las descripciones propuestas en investigaciones precedentes y se formaliza el algoritmo de transformación de reglas de negocio expresadas en SBVR a ORM2. En el Capítulo 3 se conforma el caso de estudio mostrando un ejemplo de la utilización del algoritmo de transformación para un conjunto de reglas de negocio seleccionadas a partir de una muestra del caso de estudio. Este documento culmina con las conclusiones, recomendaciones y referencias bibliográficas.

CAPÍTULO 1.

MODELADO DE REGLAS DE NEGOCIO

CAPÍTULO 1. MODELADO DE REGLAS DE NEGOCIO

La modelación de reglas de negocio es hoy, una de las tareas más importantes con las que deben tratar los desarrolladores y expertos en el tema. Pues en la mayoría de los negocios no se tiene una formalización lógica para definir sus reglas de negocio. Como resultado, trabajadores del negocio a menudo inventan las reglas sobre la marcha. Esto trae confusión, contradicción, e ineficacia operacional. La resolución de estos problemas después de ocurridos conlleva gasto de tiempo, recursos y frustración en clientes y empleados por igual. Mientras más grande es la organización, más grande el problema. La equivocación de conceptos importantes del negocio inevitablemente da por resultado una mala comunicación. En este capítulo se hará énfasis en las reglas de negocio, se dará una descripción de la notación lineal ORM2, sus componentes básicos, así como los tipos de restricciones y especificaciones que maneja la notación lineal. También se abordará el estándar para la representación de reglas y vocabularios de negocio (SBVR), sus fundamentos básicos, operadores lógicos, formalización semántica y principales problemas en la formalización.

1.1. Reglas de Negocio

Se define como “una regla que está bajo la jurisdicción del negocio”. La expresión “bajo la jurisdicción del negocio” debe interpretarse como que el negocio puede definir, revisar y eliminar reglas de negocio según sea necesario. Por ejemplo, la ley de la gravedad no es, obviamente, una regla de negocio. Igual que sucede con las reglas matemáticas.

Claramente, el término ‘regla’ tiene el sentido de ‘guía de conducta o acción’, tanto en la vida diaria como en el ámbito de los negocios.

El enfoque de reglas de negocio (BRA, acrónimo del inglés: *Business Rules Approach*) es una opción dirigida a la rápida adaptación de los sistemas de cómputo a las condiciones del cambiante mundo de los negocios. Tal enfoque modela aspectos dinámicos como reglas que las aplicaciones deben hacer cumplir mediante arquitecturas organizadas por capas. Este coloca en primer plano la captura de las políticas, regulaciones o leyes que deben observarse durante los procesos que se llevan a cabo en el negocio (Bajec, et al., 2000), representa un cambio de paradigma en el diseño y desarrollo de sistemas de negocios.

La aplicación del enfoque de reglas de negocio ha probado mejorar la efectividad, flexibilidad, y eficiencia de los sistemas de negocio, su tecnología es apropiada para cualquier tipo de

organización, y está especialmente equipado para el desarrollo en la rápida creciente industria de los servicios Web (Ronald G. Ross, 2003).

Si se examina la cuestión en mayor detalle, resulta obvio que si las reglas se van a emplear como guías de conducta o acción, deben proporcionar también los criterios precisos para juzgar y guiar la conducta o acción. En otras palabras, en el contexto de las reglas de negocio (y probablemente también en la mayoría de los dominios), las reglas se emplean como conjuntos de criterios para tomar decisiones.

Este punto es de vital importancia para todos aquellos profesionales dedicados a la creación de modelos de negocio. En la ingeniería de procesos de negocio, por ejemplo, el significado más extendido para el concepto 'regla de negocio' es como criterios para los puntos de decisión ('bifurcaciones') en modelos de procesos de negocio. Estos puntos de decisión son, habitualmente, sencillos (por ejemplo, '¿cuál es el nivel de un cliente, oro, plata o bronce?') Aunque hay casos en los que esos puntos de decisión van acompañados de criterios realmente complejos (por ejemplo, '¿debe pagarse la reclamación del seguro, debe denegarse o debe considerarse como probablemente fraudulenta?') Para estas situaciones complejas, la aplicación de técnicas de inferencia especiales puede ser de gran ayuda.

1.2. Notación Lineal ORM2 como soporte de Validación para SBVR

La idea principal de las reglas de negocio formalmente respaldada por el SBVR es la siguiente: "Las reglas se basan en hechos, y los hechos están constituidos por conceptos expresados por los términos. Los términos expresan conceptos del negocio; los hechos hacen las aseveraciones sobre estos conceptos; las reglas restringen y respaldan estos hechos" (OMG, 2008). Las notaciones de los términos y los hechos corresponden a los *noun concepts* y *verb concepts* respectivamente del SBVR (Bollen, 2009).

La consistencia de un esquema y el hecho de que alguna restricción pueda estar presente en un esquema como consecuencia implícita, son propiedades típicas de un esquema conceptual, que una vez chequeado, mejora significativamente su calidad. La verificación automática de esas propiedades sobre un esquema depende estrictamente de las posibilidades de razonamiento e inferencia sobre estas, por medio de una semántica basada en la lógica, representación del esquema en sí. Con esta meta en mente, se presenta la notación lineal ORM2 (Franconi, 2011) para cubrir el conjunto de formalizaciones lógicas propuesta por el SBVR.

Los componentes básicos que pueden ser modelados por el sistema son los *objetos*. Un objeto está compuesto por una *entidad* y un *valor* (ejemplo, un número o un string), los objetos se relacionan entre ellos a través de *predicados* de cualquier aridad. Si un objeto e está relacionado con otro f a través de un predicado p , se dice que los objetos e y f juegan un rol en el predicado p . El término *rol* es usado para referenciar la relación entre un objeto y un predicado, puesto que un objeto puede ser usado en múltiples predicados.

Recapitulando la definición de la notación lineal ORM2 desarrollada en (Franconi, 2011), muestra que la signatura Ω de la notación lineal ORM2 está compuesta por:

un conjunto \mathcal{E} de símbolos entidad;

un conjunto \mathcal{V} de símbolos valor;

un conjunto \mathcal{O} de símbolos objeto donde $\mathcal{O} = \mathcal{E} \cup \mathcal{V}$;

un conjunto \mathcal{R} de símbolos relación;

un conjunto \mathcal{A} de símbolos (atributo) rol;

un conjunto \mathcal{D} de símbolos dominio;

un conjunto Λ de conjuntos pares disjuntos de valores

una función de extensión $\Lambda_{(\cdot)}: \mathcal{D} \rightarrow \Lambda$ asociando cada símbolo dominio $D \in \mathcal{D}$ una extensión Λ_D

una relación binaria $\varrho \subseteq \mathcal{R} \times \mathcal{A}$ conectando roles con relaciones. Tomando el par relación-rol $R.a$ como el elemento atómico de la sintaxis, llamado *rol localizado*. Dado una relación R , $\varrho_R = \{R.a \mid R.a \in \varrho\}$ es el conjunto de roles localizados con respecto a R ; $\text{aridad}(R) = |\varrho_R|$ es la aridad de la relación R ;

para cada símbolo relación R , una biyección $\tau_R: \varrho_R \rightarrow [1..|\varrho_R|]$ mapea cada elemento en ϱ_R para un elemento en la sucesión finita de números naturales.

$[1..|\varrho_R|]$. Se define $\tau = \bigcup_{R \in \mathcal{R}} \tau_R$. El mapeo τ_R garantiza una correspondencia entre roles y su posición en la relación, seleccionándose libremente entre una representación basada en atributos o posicional.

1.3. Tipos de Restricciones y su especificación en la notación lineal ORM2

En este epígrafe se presentarán una serie de restricciones y cada una de sus especificaciones en la notación lineal ORM2. A continuación se describen cada una de ellas:

Conexión de rol:

$$TYPE \subseteq \varrho \times \mathcal{O}$$

Restricción de frecuencia: generaliza la restricción de singularidad, por consiguiente cada tupla ocurre un número de veces, en lugar de una única vez.

$FREQ \subseteq \wp(\varrho) \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$ – relación que indica la frecuencia de ocurrencia de un rol. Puede ser: externa (restricción de frecuencia que se le aplica como un único rol a diferentes relaciones), interna (restricción de frecuencia que se le aplica como un único rol a una única relación). La sintaxis soporta expresiones de rango, ver (Halpin, 2008).

Restricción de rol obligatorio (mandatory role): si para cada interpretación del esquema conceptual, debe ser jugado para todas las instancias de su objeto que son mencionadas en la interpretación.

$$MAND \subseteq \wp(\varrho) \times \mathcal{O}$$

Restricción de Subconjunto: puede ser aplicada sobre dos rangos de roles en dos predicados. La restricción especifica que la tupla en el rol del primer predicado tenga esa misma cantidad en el segundo predicado. Es importante que el tipo de rol del subpredicado en una restricción de subconjunto tenga correspondencia con el del superpredicado.

Restricción de Exclusión: puede ser usada para modelar una situación donde pueda ocurrir solo un valor en particular.

$R-SET_H \subseteq (\wp(\varrho) \times \wp(\varrho)) \times (\mu: \varrho \rightarrow \varrho)$, donde μ es una biyección parcial donde, para cada $\langle A, B, \mu \rangle \in R-SET_H$, $A = \{R.a | \mu(R.a) \in B\}$ y $H = \{Sub, Exc\}$. $R-SET_{Sub}$ representa restricciones de subconjunto y $R-SET_{Exc}$ representa restricciones de exclusión.

Restricción de Subtipo:

$O-SET_H \subseteq 2^{\mathcal{O}} \times \mathcal{O}$, donde $H = \{Isa, Tot, Ex\}$ – relación binaria que representa jerarquía de subtipo en los objetos.

Restricción de cardinalidad de objetos:

$O-CARD \subseteq \mathcal{O} \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$ – función parcial que define la cardinalidad de las extensiones de un objeto (ejemplo, cada población de A incluye un número de instancias que es restringida por estar un rango dado).

Restricción de cardinalidad de roles:

$R\text{-CARD} \subseteq \mathcal{R} \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$ – función parcial que define la cardinalidad de las extensiones de una relación (ejemplo, cada población de R incluye un número de instancias que es restringida por estar en un rango dado).

Objetificación: es usado para objetificar un fact type como un objeto A .

$$OBJ \subseteq \mathcal{R} \times \mathcal{O}$$

Restricción de anillo: especifica las propiedades que una relación puede poseer.

$$RING_J \subseteq \wp(\rho \times \rho), \text{ donde } J = \{Irr, Asym, Trans, Intr, Antisym, Acyclic, Sym, Ref\}$$

Restricción de valor de un objeto: es usado para enumerar el rango de valores que una entidad puede tener.

$$V\text{-VAL} : \mathcal{V} \rightarrow \wp(\Lambda_D), \text{ para algún } \Lambda_D \in \Lambda$$

Restricción modal:

$(\cdot)^\square$, $(\cdot)^\diamond$, $(\cdot)^O$ y $(\cdot)^P$ – representan los operadores modales de necesidad, permisión, obligación y permisión respectivamente, que son aplicables a todas las relaciones en Ω (ejemplo, OBJ^\square).

Restricciones derivadas:

Restricciones de singularidad

$$FREQ(\{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, (0,1))$$

Restricción de valor de un rol:

$TYPE(R.a, V^*)$, donde V^* es un nuevo valor

$$V\text{-VAL}(V^*) \subseteq \{v_1^D, \dots, v_n^D\}$$

Restricción de igualdad:

$$R\text{-SET}_{Sub} \left(\begin{array}{l} \{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, \\ \{S^1.b_{11}, \dots, S^1.b_{1v}, \dots, S^q.a_{q1}, \dots, S^q.a_{qw}\}, \mu \end{array} \right)$$

$$R\text{-SET}_{Sub} \left(\begin{array}{l} \{S^1.b_{11}, \dots, S^1.b_{1v}, \dots, S^q.a_{q1}, \dots, S^q.a_{qw}\}, \\ \{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, \mu^- \end{array} \right)$$

Restricción de o-exclusivo:

$$MAND(\{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, O)$$

$$R\text{-SET}_{Exc}(\{R^1.a_{11}, \dots, R^1.a_{1n}\}, \{R^2.a_{21}, \dots, R^2.a_{2n}\}, \mu^1)$$

$R\text{-SET}_{Exc}(\{R^1.a_{11}, \dots, R^1.a_{1n}\}, \{R^3.a_{31}, \dots, R^3.a_{3n}\}, \mu^2) \dots$

$R\text{-SET}_{Exc}(\{R^{k-1}.a_{k-11}, \dots, R^{k-1}.a_{k-1n}\}, \{R^k.a_{k1}, \dots, R^k.a_{kn}\}, \mu^k)$

Dada la sintaxis de la notación lineal ORM2, en (Solomakhin, 2011) se definió un formato de entrada implementable, representación del conjunto de elementos de la signatura Ω que se muestra en la Tabla 1.1.

Codificación FOL para ORM2

El SBVR provee una formalización lógica capaz de representar las reglas de negocio, por lo que nos apoyamos en su semántica de FOL (Lógica de primer orden) y basados en una signatura Ω_{FOL} , mostramos su equivalencia con la sintaxis lineal, como se muestra a continuación:

Tabla1-1Notación lineal ORM2

Sintaxis lineal ORM2	Formato de entrada	Ejemplos
Conjunto \mathcal{E} de símbolos entidad	$ENTITYTYPES: \{Ent [, Ent] *\}$	$ENTITYTYPES: \{Paciente, Receptor\}$
Conjunto \mathcal{V} de símbolos valor	$VALUETYPES: \{Val [, Val] *\}$	$VALUETYPES: \{Paciente_Nr, Receptor_Nombre\}$
Conjunto \mathcal{R} de símbolos relación	$RELATIONS: \{Rel [, Rel] *\}$	$RELATIONS: \{asociadoA, debeComenzar\}$
Conjunto \mathcal{A} de símbolos rol	$ATTRIBUTES: \{atr [, atr] *\}$	$ATTRIBUTES: \{paciente, receptor\}$
Conjunto \mathcal{D} de símbolos dominio	$DOMAIN: \{DomainElem [, DomainElem] *\}$	$DOMAIN: \{hospital\}$
<p>Las entidades y valores empiezan con mayúscula, las relaciones, los roles y los dominios pueden empezar con cualquier tipo de letra, pueden contener letras, números y los siguientes símbolos: ‘_’ y ‘-’. El conjunto de roles, dominios, relaciones, entidades y valores deben ser disjuntos.</p>		
Relación binaria $\rho \subseteq \mathcal{R} \times \mathcal{A}$	$LOC-ROLES: \{Rel. atr [, Rel. atr] *\}$	$LOC-ROLES: \{probadoPor.funcion, probadoPor.metodo\}$
Bijección $\tau_R: \mathcal{Q}_R \rightarrow [1..[q_R]]$	$LOC-ROLES-INDEX(Rel. atr, i)$ $LOC-ROLES-INDEX: \{(Rel. atr, i)[, (Rel. atr, i) *\}$	$LOC-ROLES-INDEX(tieneCalidad.calidad, 2)$

Sintaxis lineal ORM2	Formato de entrada	Ejemplos
$TYPE \subseteq \varrho \times \mathcal{O}$	$TYPE(Rel. atr, Obj)$	$TYPE(tieneCosto.metodo, Metodo)$
$FREQ \subseteq \wp(\varrho) \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$	$FREQ(\{Rel. atr [, Rel. atr] *\}, (i, j))$, donde $i \geq 0$ y $j > 0$ o $j = inf$ o $j = INF$, correspondiente a $j = +\infty$	$FREQ(\{tieneCalidad.metodo\}, (1,1))$
$MAND \subseteq \wp(\varrho) \times \mathcal{O}$	$MAND(\{Rel. atr [, Rel. atr] *\}, Obj)$	$MAND(\{esIdentificadoPor.metodo\}, Metodo)$
$R-SET_H \subseteq (\wp(\varrho) \times \wp(\varrho)) \times (\mu: \varrho \rightarrow \varrho)$	$R-SETh$ $\left(\begin{array}{l} \{Rel. atr [, Rel. atr] *\}, \\ \{Rel. atr [, Rel. atr] *\}, \\ \{Rel. atr, Rel. atr [, Rel. atr, Rel. atr] *\} \end{array} \right)$, donde $h = [sub exc]$	$R-SETsub$ $\left(\begin{array}{l} \{estaEn.curso\}, \\ \{activoEn.cursoActivo\}, \\ \{estaEn.curso, activoEn.cursoActivo\} \end{array} \right)$
$O-SET_H \subseteq 2^{\mathcal{O}} \times \mathcal{O}$, donde $H = \{Isa, Tot, Ex\}$	$O-SETh(\{Obj [, Obj] *\}, Obj)$, donde $h = [isa tot ex]$	$O-SETtot(\{MetodoFisico, MetodoVirtual\}, Metodo)$
$O-CARD \subseteq \mathcal{O} \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$	$O-CARD(Obj) = (i, j)$, donde $i \geq 0$ y $j > 0$ o $j = inf$ o $j = INF$, correspondiente a $j = +\infty$	$O-CARD(CursoActivo) = (1,50)$
$R-CARD \subseteq \mathcal{R} \times (\mathbb{N} \times (\mathbb{N} \cup \{\infty\}))$	$R-CARD(Rel) = (i, j)$, donde $i \geq 0$ y $j > 0$ o $j = inf$ o $j = INF$, correspondiente a $j = +\infty$	$R-CARD(tieneCalidad) = (1, INF)$
$OBJ \subseteq \mathcal{R} \times \mathcal{O}$	$OBJ(Rel, Obj)$	$OBJ(tienePrerequisito, O_Prerequisito)$

Sintaxis lineal ORM2	Formato de entrada	Ejemplos
$RING_J \subseteq \wp(\varrho \times \varrho)$, donde $J = \left\{ \begin{array}{l} Irr, Asym, Trans, \\ Intr, Antisym, Acyclic, \\ Sym, Ref \end{array} \right\}$	$RING_j(Rel. atr, Rel. atr)$, donde $j = [irr asym trans intr antisym acyclic] aym ref$	$RING_{irr} \left(\begin{array}{l} tienePrerequisito.materia, \\ tienePrerequisito.object \end{array} \right)$
$V-Val : \mathcal{V} \rightarrow \wp(\Lambda_D)$, para algún $\Lambda_D \in \Lambda$	$V-VAL(val) = \left\{ \begin{array}{l} (value..value) (..value) (value..) \\ value..value ..value \end{array} \right\}$ $V-VAL(val) = \{value [, value] *\}$ $V-VAL(val) = \{< xsd: datatype >\}$ Value puede ser un entero, un float o una constante string encerrado entre ". Las restricciones de rangos son definidos solo para números. Las restricciones de rangos pueden ser inclusivas [] o exclusivas (.). Los rangos ilimitados (ejemplo, (value..)) no pueden ser inclusivos. Cuando los extremos no se expresan explícitamente son incluidos. xds:datatype es cualquier tipo de dato del esquema XML.	$V-VAL(Curso_Codigo) = \{[101..399]\}$ $V-VAL(Estudiante_Nr) = \{< xsd: decimal >\}$
$(\cdot)^{\square}$, $(\cdot)^{\circ}$, $(\cdot)^O$ y $(\cdot)^P$	$\backslash BOX\{\cdot\}$, $\backslash DIA\{\cdot\}$, $\backslash OB\{\cdot\}$ y $PM\{\cdot\}$ respectivamente	$\backslash OB \left\{ MAND \left(\begin{array}{l} esIdentificadoPor.metodo, \\ Metodo \end{array} \right) \right\}$

- E_1, E_2, \dots, E_n predicados unarios para entidades;
- V_1, V_2, \dots, V_m predicados unarios para valores;
- D_1, D_2, \dots, D_l predicados unarios para dominios;
- R_1, R_2, \dots, R_k predicados n-arios para relaciones;
- un conjunto contable de constantes d_1, d_2, \dots ;
- un conjunto $ID^2, \dots, ID^{n_{max}}$ de funciones, $n_{max} = \max\{|Q_R| | R \in \mathcal{R}\}$.

La codificación de FOL de la semántica ORM2 introducida anteriormente es la siguiente:

$$\forall x \left(E_i(x) \rightarrow \neg(D_1(x) \vee \dots \vee D_l(x)) \right), \text{ para } 1 \leq i \leq n \quad (2.1)$$

$$\forall x \left(V_i(x) \rightarrow D_j(x) \right), \text{ para } 1 \leq i \leq m \quad (2.2)$$

$$\forall x \left(D_i(x) \leftrightarrow (x = d_1 \vee x = d_2 \vee \dots) \right), \text{ para todo } d_i \in \Lambda_{D_i} \quad (2.3)$$

$$\forall x_1, \dots, x_n, z_1, \dots, z_n \left(ID(\bar{x}) = ID(\bar{z}) \leftrightarrow \bar{x} = \bar{z} \right), \text{ para } n = 1, \dots, n_{max} \quad (2.4)$$

El anterior conjunto de axiomas se necesita con el objetivo de forzar la interpretación de los símbolos en la base de conocimiento de FOL a que sea correcta. En particular (2.1) fuerza la interpretación de cada entidad a ser disjunta de la interpretación de los dominios; el axioma (2.2) muestra que objetos en la interpretación de un valor debe estar también en un dominio específico; el axioma (2.3) fuerza a la interpretación de un dominio a estar entre el conjunto de valores predefinidos por $\Lambda_{(\cdot)}$; mientras que el axioma (2.4) captura la naturaleza inyectiva de cada función ID y el hecho de que tuplas de diferente longitud nunca coincidirán en el mismo identificador.

Representación de una *conexión de rol*

$$\forall x_1 \dots x_{\tau(R.a)} \dots x_n \left(R(x_1, \dots, x_{\tau(R.a)}, \dots, x_n) \rightarrow O(x_{\tau(R.a)}) \right), \text{ su transformación equivalente}$$

es $TYPE(R.a, O)$.

Representación de una restricción de frecuencia

$$\forall \bar{y} \left[\exists \bar{x}^1 \dots \bar{x}^k \left(\bigwedge_{j=1}^k R^j(\bar{x}^j) \wedge \bigwedge_{i1=1}^n (x_{\tau(R^1.a_{1i1})}^1 = y_{1i1}) \wedge \dots \wedge \bigwedge_{ik=1}^m (x_{\tau(R^1.a_{1ik})}^1 = y_{1ik}) \right) \right]$$

$$\rightarrow \exists^{\min..max} \bar{y} \left[\exists \bar{x}^1 \dots \bar{x}^k \left(\bigwedge_{j=1}^k R^j(\bar{x}^j) \wedge \bigwedge_{i1=1}^n (x_{\tau(R^1.a_{1i1})}^1 = y_{1i1}) \wedge \dots \wedge \bigwedge_{ik=1}^m (x_{\tau(R^1.a_{1ik})}^1 = y_{1ik}) \right) \right]$$

su transformación equivalente es

$FREQ(\{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, (min, max))$ donde $R^i.a_{ix} \in \mathcal{Q}_{R^i}$ para cualquier $R^i \in R$.

Representación de una restricción de rol obligatorio

$$\forall y \left[O(y) \rightarrow \bigvee_{l=1}^n \exists \bar{z}^l (R^l(\bar{z}^l) \wedge (z_{\tau(R^l.a_{li})}^l = y)) \vee \dots \vee \bigvee_{j=1}^m \exists \bar{z}^j (R^k(\bar{z}^j) \wedge (z_{\tau(R^k.a_{kj})}^j = y)) \right]$$

su transformación equivalente es $MAND(\{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, O)$.

Representación de una restricción de subtipo sobre relaciones, de tipo subconjunto

$$\forall \bar{y} \left[\exists \bar{x}^1 \dots \bar{x}^k \left(\bigwedge_{j=1}^k R^j(\bar{x}^j) \wedge \bigwedge_{i1=1}^n (x_{\tau(R^1.a_{1i1})}^1 = y_{1i1}) \wedge \dots \wedge \bigwedge_{ik=1}^m (x_{\tau(R^1.a_{1ik})}^1 = y_{1ik}) \right) \right]$$

$$\rightarrow \exists \bar{z}^1 \dots \bar{z}^q \left(\bigwedge_{i=1}^q S^i(\bar{z}^i) \wedge \bigwedge_{i1=1}^n (z_{\tau(\mu(R^1.a_{1i1})}^{f\mu(1i1)})} = y_{1i1}) \wedge \dots \wedge \bigwedge_{ik=1}^m (z_{\tau(\mu(R^1.a_{1ik})}^{f\mu(1ik)})} = y_{1ik}) \right)$$

y de tipo exclusión

$$\forall \bar{y} \left[\exists \bar{x}^1 \dots \bar{x}^k \left(\bigwedge_{j=1}^k R^j(\bar{x}^j) \wedge \bigwedge_{i1=1}^n (x_{\tau(R^1.a_{1i1})}^1 = y_{1i1}) \wedge \dots \wedge \bigwedge_{ik=1}^m (x_{\tau(R^1.a_{1ik})}^1 = y_{1ik}) \right) \right]$$

$$\rightarrow \neg \left(\exists \bar{z}^1 \dots \bar{z}^q \left(\bigwedge_{i=1}^q S^i(\bar{z}^i) \wedge \bigwedge_{i1=1}^n (z_{\tau(\mu(R^1.a_{1i1})}^{f\mu(1i1)})} = y_{1i1}) \wedge \dots \wedge \bigwedge_{ik=1}^m (z_{\tau(\mu(R^1.a_{1ik})}^{f\mu(1ik)})} = y_{1ik}) \right) \right)$$

su transformación equivalente es $R\text{-SETH}(\{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}, \{S^1.b_{11}, \dots, S^1.b_{1v}, \dots, S^q.a_{q1}, \dots, S^q.a_{qw}\}, \mu)$

donde $H = \{sub, exc\}$, $\mathcal{Q}^{CA} = \{R^1.a_{11}, \dots, R^1.a_{1n}, \dots, R^k.a_{k1}, \dots, R^k.a_{km}\}$, $\mathcal{Q}^{CB} =$

$\{S^1.b_{11}, \dots, S^1.b_{1v}, \dots, S^q.a_{q1}, \dots, S^q.a_{qw}\}$, μ es una biyección parcial para cualquier $\langle \varrho^{CA}, \varrho^{CB}, \mu \rangle \in \mathbf{R}\text{-SETH}$ donde $\varrho^{CA} = \{R.a \mid \mu(R.a) \in \varrho^{CB}\}$ y $f_{\mu(xy)} = z_{\text{SSI}} \mu(R^x.a_{xy} \in \varrho^{SZ})$.

Representación de una restricción de jerarquía:

de tipo parcial $\forall y (O_i(y) \rightarrow O(y))$, para todo $i = \overline{1..n}$,

de tipo total $\left\{ \begin{array}{l} \forall y (O_i(y) \rightarrow O(y)) \\ \forall y (O(y) \rightarrow O_1(y) \vee \dots \vee O_n(y)) \end{array} \right.$, para todo $i = \overline{1..n}$,

y de tipo exclusiva $\left\{ \begin{array}{l} \forall y (O_1(y) \rightarrow O(y) \wedge \neg O_2(y) \wedge \dots \wedge \neg O_n(y)) \\ \forall y (O_2(y) \rightarrow O(y) \wedge \neg O_3(y) \wedge \dots \wedge \neg O_{n-1}(y)) \\ \dots \\ \forall y (O_{n-1}(y) \rightarrow O(y) \wedge \neg O_1(y)) \\ \forall y (O_n(y) \rightarrow O(y)) \end{array} \right.$

su transformación equivalente es $\mathbf{O}\text{-SETH}(\{O_1, \dots, O_n\}, O)$ donde $H = \{isa, tot, ex\}$.

Representación de una restricción de cardinalidad de objeto $\exists^{\geq \min} y (O(y)) \wedge$

$\exists^{\leq \max} y (O(y))$, su transformación equivalente es $\mathbf{O}\text{-CARD}(O) = (min, max)$.

Representación de una restricción de cardinalidad de rol

$\exists^{\geq \min} x_{\tau(R.a)} (R(x_1, \dots, x_{\tau(R.a)}, \dots, x_n)) \wedge \exists^{\leq \max} x_{\tau(R.a)} (R(x_1, \dots, x_{\tau(R.a)}, \dots, x_n))$

su transformación equivalente es $\mathbf{R}\text{-CARD}(R.a) = (min, max)$.

Representación de una objetificación $\forall x (O(x) \leftrightarrow \exists \bar{y} (R(\bar{y}) \wedge ID^{|\varrho^R|}(\bar{y}) = x))$, su transformación equivalente es $\mathbf{OBJ}(R, O)$.

1.4. Estándar para la representación de reglas y vocabularios de negocio

En los modernos procesos de negocio los analistas determinan y administran las reglas, los sistemas de administración de reglas de negocio son empleados para separar la lógica del negocio del código de la aplicación. Separar la lógica tiene como principal fin que las aplicaciones posean una mayor adaptabilidad, de manera que pueda extenderse su vida útil ante

los cambios que se producen en la política de las empresas que le dan soporte a dichas aplicaciones. SBVR provee una forma de documentar la semántica del vocabulario y las reglas especificados en un lenguaje familiar a los usuarios del negocio.

1.4.1. Fundamentos Básicos de SBVR

Definición 1. Un *noun concept* es "un concepto" el cuál puede clasificarse en las siguientes formas: un *object type*, un *individual concept* o un *fact type role*. Un *object type* es "un *noun concept* que clasifica cosas en base a sus propiedades comunes". Un *individual concept* es "un concepto que corresponde con solamente un objeto" (OMG, 2008).

Un *fact type role* es "un *noun concept* que corresponde con objetos basados en su rol, asumiendo una función o siendo usado en alguna situación".

Definición 2. Un *verb concept* (o un *fact type*) representa la noción de las relaciones. Un *fact type* puede tener uno (característica), dos (binario) o más *fact type roles* (OMG, 2008).

El estándar de SBVR define dos tipos de reglas de negocio: reglas estructurales (de definición) y reglas operativas (conductuales).

Definición 3. Las reglas estructurales, son reglas que permiten definir cómo se organiza o estructura el negocio y que nunca pueden ser violadas de forma directa. Simplemente definen características que permiten categorizar (u organizar o estructurar) las entidades que intervienen en el negocio (definen características de los *noun concept* y restricciones sobre los *verb concepts*).

La mala definición o aplicación errónea de este tipo de reglas conduce a resultados erróneos o incoherentes pero no a sanciones o penalizaciones como sucede cuando se viola una regla operativa. A su vez, se distinguen dos tipos de reglas estructurales: las reglas de inferencia, aquellas que permiten, como su nombre indica, inferir algún nuevo dato, y las reglas de cálculo, que indican las operaciones matemáticas necesarias para obtener un dato numérico determinado.

Una regla de negocio estructural puede ser expresada usando modalidades como son: necesidad, imposibilidad o posibilidad restringida.

Necesidad y posibilidad:

Una “**necesidad**” o “**posibilidad**” suelen ser complementarias a una definición. El campo de “**necesidad**” se utiliza para afirmar algo que es necesariamente cierto. El campo de “**posibilidad**”, explica que algo es una posibilidad que no se impide por definición. La frase

clave ‘es necesario que’ puede ser omitida en una declaración de una regla estructural titulada “**Necesidad**”, porque es implícito en el título. Estos son algunos ejemplos, dos de necesidad y una de posibilidad.

Representación

Necesidad: **Todo examen físico es un examen.**

Necesidad: **Todo examen realizado tiene que ser asociado a un paciente.**

Posibilidad: **El resultado de los análisis de los pacientes es desechado después del trasplante renal.**

Las definiciones expresan características que son necesarias y suficientes para distinguir las cosas denotadas por un concepto. A veces hay necesidades más allá de lo que es suficiente. El campo de “necesidad” se utiliza para indicar dichas necesidades.

Esquema de Referencia:

Se utiliza para indicar cómo las cosas denotadas por el término se pueden distinguir unos de otros basados en uno o más hechos acerca de las cosas. Un esquema de referencia se expresa por referencia a al menos un rol de un tipo de hecho binario e indicando si una referencia implica una sola instancia de la función o si se trata de la extensión de instancias relacionadas. Un artículo (‘**el**’, ‘**un**’, o ‘**la**’) indica un uso simple del rol en el que un solo caso se utiliza en la referencia. El artículo “la” es solamente adecuado cuando puede haber como máximo una instancia del rol. Las palabras ‘el conjunto de’ indican que la extensión es utilizada. La palabra ‘y’ se utiliza para conectar las expresiones de uso de múltiples roles para un esquema de referencia.

Definición 4. Las reglas operativas (deónticas), son aquellas que son usadas para describir procesos de negocio en una organización y no pueden ser ignoradas o violadas por algún agente del negocio. Por ejemplo, la regla *todo usuario debe tener acceso a su cuenta* es una regla operativa que puede ser contravenida en un momento dado. Si se viola una regla de este tipo se producirá una penalización. En el ejemplo dado, supongamos que el usuario emplea esa cuenta para conectarse a Internet y paga por ella. Si en un momento dado se le deniega el acceso, es posible que haya que abonarle una cierta cantidad por no haber podido emplear un servicio que está pagando.

Una regla de negocio operativa puede ser expresada usando modalidades como son: obligación, prohibición o permiso restringido.

1.4.2. Formalización Semántica de SBVR

Las formalizaciones semánticas (OMG, 2008), no son expresiones ni declaraciones, son estructuras que componen significados. Hay dos tipos de formalizaciones semánticas, el primer tipo son formulaciones lógicas y proposiciones estructuradas, ambas pueden ser simples o complejas. Las especializaciones de este tipo están dadas por varias operaciones lógicas, cuantificadores, formulaciones atómicas basadas en *fact types* y otras formulaciones para propósitos especiales como objetificaciones y nominalizaciones. El segundo tipo son las proyecciones, estas formulan definiciones, agregaciones y preguntas.

Una instancia es una formulación lógica que habilita un estado o evento indicado proposicionalmente para ser el sujeto u objeto de otra proposición. Una formulación abarcadora puede relacionar un estado o evento indicado usando objetificación en puntos en el tiempo, periodos u otros estados o eventos (posiblemente también identificados usando objetificación) con respecto al tiempo (ejemplo, ocurrido antes o después). Las relaciones específicas de interés pueden ser definidas como *fact types*.

El SBVR trata el tiempo en relación a los estados y eventos permitiendo relaciones temporales definidas genérica y ortogonalmente para muchos *fact types* cuyas extensiones cambian con el tiempo.

Una nominalización proposicional es similar a la objetificación. Es un tipo de formulación lógica que estructura el significado representado por una mención de una declaración o proposición opuesta al uso de esta. Otros tipos similares de formulaciones estructuran significados representados por mención de conceptos, preguntas y respuestas. Además las reglas con cambios a menudo envuelven *noun concept* nominalizados, que son formulaciones especiales para que un concepto sea sujeto u objeto de una preposición, en muchos, de la misma manera en que una proposición nominalizada permite que una proposición sea sujeto u objeto.

1.4.3. Problemas en la formalización de SBVR

El principal problema en la formalización de SBVR para reglas de negocio es que las reglas estructurales y operacionales son consideradas de manera independiente, sin tomar en cuenta posibles inconsistencias entre ellas. Por ejemplo, algunas reglas pueden ser declaradas como una formulación estructural de necesidad y al mismo tiempo la negación de la formulación lógica usada en la regla puede participar en alguna regla deóntica, consideremos este ejemplo:

Es necesario comer.

Es permitido no comer.

Aunque la notificación de permiso no significa quitar ningún grado de libertad, este conjunto de reglas es intuitivamente incierta, sin embargo no siempre es el caso cuando tratamos con reglas con cuerpos contradictorios. Por ejemplo:

Es obligatorio no tomar bebidas alcohólicas.

Es posible tomar bebidas alcohólicas.

Este conjunto de reglas es claramente consistente, ciertamente, es físicamente posible tomar bebidas alcohólicas pero puede ser prohibido hacerlo en áreas públicas.

Además, la semántica basada en etiquetas, propuesta por el SBVR, no es capaz de capturar las dependencias y contradicciones entre las reglas de similar naturaleza. Sin embargo, la tarea de detectar posibles contradicciones e inconsistencias con el modelo es muy importante para el modelado del negocio. Si no hay procedimientos específicos para definir si el conjunto de reglas es consistente, es altamente probable que el modelo resultante sea inútil.

1.4.4. Operadores lógicos usados en SBVR

SBVR posee un fuerte basamento teórico sobre la lógica de predicados de primer orden con algunas extensiones a la lógica modal, esto lo hace coherente con lo planteado en el artículo 5 del Manifiesto de Reglas de Negocio (BRG, 2003): *“Las lógicas formales, como la lógica de predicados, son fundamentales para la expresión formal de reglas en términos de negocio, así como para las tecnologías que implementan dichas reglas.”*

A continuación se muestran los cuantificadores y operadores modales o proposicionales de los que hace uso. Las letras „ n “ y „ m “ representan números enteros. Las letras „ p “ y „ q “ representan expresiones de proposiciones.

Tabla 1-2 Cuantificadores de la lógica de predicados

Cada	cuantificación universal
Algún	cuantificación existencial
al menos uno	cuantificación existencial
al menos n	cuantificación al-menos- n
a lo sumo uno	cuantificación a-lo-sumo-uno
a lo sumo n	cuantificación al-lo-sumo- n
exactamente uno	cuantificación exactamente-uno
exactamente n	cuantificación exactamente- n
al menos n y a lo sumo m	cuantificación de rango numérico
más que uno	cuantificación al-menos- n con $n=2$

A continuación se muestra el conjunto de operadores de la lógica proposicional que son manejados por SBVR.

Tabla 1-3 Operadores lógicos proposicionales

no es el caso que p	negación lógica
p y q	conjunción
p o q	disyunción
p o q pero no ambos	disyunción exclusiva
si p entonces q	Implicación
q si p	Implicación
p si y solo si q	Equivalencia
no ambos p y q	formulación nand
ni p ni q	formulación nor
p tanto o no q	formulación tanto-o-no

Este último operador está compuesto por dos operandos lógicos, o sea dos proposiciones, donde „ p “ se considera un consecuente y „ q “ un inconsecuente. O sea, el valor lógico de „ q “ no altera el resultado de aplicar el operador lógico, mientras que el valor de „ p “ se obtiene como resultado de aplicar el operador lógico. En general los operadores de implicación poseen al menos un antecedente (premisa) y al menos un consecuente (conclusión). Cuando un sujeto se repite al usar „y“ o „o“ el sujeto repetido puede elidirse. Por ejemplo, la declaración “Un conductor tiene una licencia y un conductor respeta el tráfico”, puede abreviarse a: “Un conductor tiene una licencia y respeta el tráfico”. De manera similar, pueden elidirse un sujeto y un verbo repetidos. Por ejemplo, la declaración, “Un conductor tiene una licencia y un conductor tiene un carro”, puede abreviarse a: “Un conductor tiene una licencia y un carro”.

Tabla 1-4 Operadores Modales

es obligatorio que p	formulación de obligación	Operativos
es prohibido que p	formulación de obligación dentro de una negación lógica	
es permitido que p	formulación de permisibilidad	
es necesario que p	formulación de necesidad	Estructurales
es imposible que p	formulación de necesidad dentro de una negación lógica	
es posible que p	formulación de posibilidad	

1.5. Conclusiones parciales del capítulo

En este capítulo se realiza un acercamiento a la modelación de reglas de negocio, para lograr una comprensión adecuada de su aplicación y beneficios desde la perspectiva de negocio, para la misma se dispone de un teoría profunda con basamentos sólidos los cuales demuestran que las reglas de negocio juegan un papel vital en el cumplimiento de las metas organizacionales. Se hace énfasis en el estándar SBVR el cual propone una vía para la documentación de la semántica de vocabularios y reglas del negocio. Se describen la notación lineal ORM2, sus principales elementos así como cada una de las restricciones que propone la misma para las transformaciones de reglas de negocio escritas sobre SBVR.

CAPÍTULO 2.

FORMALIZACIÓN DE UN ALGORITMO PARA LA TRANSFORMACIÓN DE REGLAS DE NEGOCIO EXPRESADAS EN SBVR A ORM2

CAPÍTULO 2. FORMALIZACIÓN DE UN ALGORITMO PARA LA TRANSFORMACIÓN DE REGLAS DE NEGOCIO EXPRESADAS EN SBVR A ORM2

Basados en el conjunto de formalizaciones semánticas del SBVR se propone la transformación hacia la notación lineal ORM2 (Franconi, 2011) soportada sobre una semántica FOL propuesta por primera vez en (Halpin, 1989), después revisada y extendida en varios trabajos.

La transformación asignará notaciones de acuerdo los elementos mostrados en las formalizaciones semánticas de las reglas, basándonos en algunos fragmentos se mostrarán detalles importantes a tener en cuenta en la misma.

2.1. Transformación de Reglas de Negocio

En este epígrafe se presentan una serie de transformaciones de reglas de negocio expresadas en SBVR al formato ORM2. A continuación se presentan las siguientes transformaciones:

1: Implicación

Hay dos casos de formalización para una implicación, *si p entonces q* o *q si p*, las dos son muy semejantes solo cambia el orden del antecedente y el consecuente, como ejemplo se muestra el segundo caso a través de un *fact type* unario y uno binario respectivamente,

... El consecuente de la implicación es una formulación atómica.

... La formulación atómica es basada en el *fact type* '< fact type₁ >'.

... El rol '< fact type role₁ >' está enlazado a la < cardinalidad₁ > de la variable.

... El antecedente de la implicación es un cuantificador existencial.

... El cuantificador existencial introduce una < cardinalidad₂ > de la variable.

... La < cardinalidad₂ > variable varía sobre el concepto '< fact type role₂ >'.

... La cuantificación existencial se extiende sobre una formulación atómica.

... La formulación atómica es basada en el *fact type* '< fact type₂ >'.

... El rol '< fact type role₂ >' está enlazado a la < cardinalidad₂ > de la variable.

... El rol '< fact type role₁ >' está enlazado a la < cardinalidad₁ > de la variable.

LOC-ROLES-INDEX(Rel₁#.var_{card1}, 1)

$TYPE(Rel_1\#.var_{card1}, UppercaseWord (var_{card1}))$

$TYPE(Rel_1\#.atr, Val) \quad con = Rel_1\#.var$

var_{card1}

$LOC-ROLES-INDEX(Rel_2\#.var_{card2}, 1)$

$TYPE(Rel_2\#.var_{card2}, UppercaseWord(var_{card2}))$

$LOC-ROLES-INDEX(Rel_2\#.var_{card1}, 2)$

$TYPE(Rel_2\#.var_{card1}, UppercaseWord(var_{card1}))var_{ant} = Rel_2\#.var_{card2}$

$R-SETsub(\{var_{con}\}, \{var_{ant}\}, \{var_{con}, var_{ant}\})$

2: Disyunción

2.1: con respecto a entidades

... El primer operando lógico de la disyunción es una formulación de instanciación.

... La formulación de instanciación considera el concepto '< fact type role₁ >'.

... La formulación de instanciación enlaza el concepto '< fact type role₂ >'.

... El segundo operando lógico de la disyunción es una formulación de instanciación.

... La formulación de instanciación considera el concepto '< fact type role₁ >'.

... La formulación de instanciación enlaza el concepto '< fact type role₃ >'.

$O-SETexc(\{Obj_2, Obj_3\}, Obj_1)$

donde los *Obj* representan los < fact type role > según sus índices.

Si en la regla intervienen elementos de frecuencia la restricción puede indicar totalidad,

$O-SET_{tot}(\{Obj_2, Obj_3\}, Obj_1)$

2.2: con respecto a relaciones

Se mostrará a través de un fact type binario:

... El primer operando lógico de la disyunción es una formulación atómica.

... La formulación atómica es basada en el fact type '< fact type₁ >'.

... El rol '< fact type role₁ >' está enlazado a la < cardinalidad₁ > de la variable.

... El rol '< fact type role₂ >' está enlazado a la < cardinalidad₂ > de la variable.

. . . El segundo operando lógico de la disyunción es una formulación atómica.

. . . . La formulación atómica es basada en el fact type '< fact type₂ >'.

. El rol '< fact type role₁ >' está enlazado a la < cardinalidad₁ > de la variable.

. El rol '< fact type role₂ >' está enlazado a la < cardinalidad₂ > de la variable.

LOC-ROLES-INDEX(Rel₁#. var_{card1}, 1)

TYPE(Rel₁#. var_{card1}, UppercaseWord(var_{card1}))

LOC-ROLES-INDEX(Rel₁#. var_{card2}, 2)

TYPE(Rel₁#. var_{card2}, UppercaseWord(var_{card2}))

LOC-ROLES-INDEX(Rel₂#. var_{card1}, 1)

TYPE(Rel₂#. var_{card1}, UppercaseWord(var_{card1}))

LOC-ROLES-INDEX(Rel₂#. var_{card2}, 2)

TYPE(Rel₂#. var_{card2}, UppercaseWord(var_{card2}))

R-SET_{exc}({Rel₁#. var_{card1}}, {Rel₂#. var_{card1}}, {Rel₁#. var_{card1}, Rel₂#. var_{card1}})

3: Roles de mandamiento

(La proposición es significado de/la declaración es (formulada/formalizada) por) una cuantificación universal.

⋮

. La cuantificación universal se extiende sobre una cuantificación existencial.

⋮

. . La cuantificación existencial se extiende sobre una formulación atómica.

Estas formalizaciones sobre las reglas indican que el rol principal de la formulación atómica es de mandamiento, que a su vez, por defecto implica que sea de necesidad, un ejemplo concreto sobre una formalización atómica sería:

MAND(Rel#. atr, UppercaseWord(atr))

4: Restricciones de frecuencia

Las restricciones de cuantificación como-máximo-uno, como-máximo-n, exactamente-uno, exactamente-n y rango numérico son representadas con la sintaxis *FREQ*, mientras que la cuantificación al-menos-n se representa con *R-CARD*, a continuación se muestran:

. . . *La cuantificación como-máximo-n tiene cardinalidad máximan.*

. . . *La cuantificación como-máximo-n introduce una < cardinalidad > de la variable.*

. . . . *La < cardinalidad > variable varía sobre el concepto '< fact type role >'.*

. . . *La cuantificación como-máximo-n se extiende sobre una formulación atómica.*

. . . . *La formulación atómica es basada en el fact type '< fact type >'.*

FREQ({*Rel#. var_{card}*}, (0, *n*))

En caso de ser al-menos-n la transformación sería *R-CARD*(*Rel#*) = (*n*, *INF*)

5: Formulaciones modales

Las formulaciones modales son basadas en las modalidades de necesidad, posibilidad, permisión y obligación. Un ejemplo de una formulación de obligación es la siguiente:

La regla es una proposición significado de una formulación de obligación.

La declaración es formulada por una formulación de obligación.

Las formulaciones modales también indican roles de mandamiento, excepto la modalidad de posibilidad, e influyen sobre las restricciones de frecuencias, de subconjunto, de exclusión y de cardinalidad de los roles, por ejemplo:

$\backslash OB \{ MAND(Rel\#. atr, UppercaseWord(atr)) \}$

6: Variable restringida por una proposición de nominalización

Se mostrará la transformación a través de un fact type binario como ejemplo:

. . *La < cardinalidad₁ > de la variable varía sobre el concepto '< fact type role₁ >'.*

. . . *La < cardinalidad₂ > de la variable varía sobre el concepto 'proposición'.*

. . . . *La < cardinalidad₂ > de la variable está restringida por una proposición de nominalización.*

. *La proposición de nominalización enlaza a la < cardinalidad₂ > de la variable.*

. *La < cardinalidad₃ > variable varía sobre el concepto '< fact type role₂ >'.*

⋮

... .La formulación atómica está basada en el fact type '< fact type >'.

... .El rol '< fact type role₁ >' está enlazado a la < cardinalidad₁ > de la variable.

... .El rol '< fact type role₂ >' está enlazado a la < cardinalidad₃ > de la variable.

una proposición de nominalización objetifica la formulación atómica y hereda de proposición,

LOC-ROLES-INDEX(Rel#. var_{card1}, 1)

TYPE(Rel#. var_{card1}, UppercaseWord(var_{card1}))

LOC-ROLES-INDEX(Rel#. var_{card3}, 1)

TYPE(Rel#. var_{card3}, UppercaseWord(var_{card3}))

OBJ(Rel#, Obj) *O-SET*_{isa}({Obj}, Proposicion)

7: Variable restringida por una objetificación

La objetificación se ejemplificará aplicándose a un fact type unario:

... .La < cardinalidad₂ > de la variable es restringida por una objetificación.

... .La objetificación enlaza a la < cardinalidad₂ > de la variable.

... .La objetificación considera una formulación atómica.

... .La formulación atómica es basada en el fact type '< fact type >'.

... .El rol '< fact type role >' está unido a la < cardinalidad₁ > de la variable.

LOC-ROLES-INDEX(Rel#. var_{card1}, 1)

TYPE(Rel#. var_{card1}, UppercaseWord(var_{card1}))

TYPE(Rel#. atr, Val) *OBJ*(Rel#, Obj) var₃ = Obj

Para concluir, no se muestran transformaciones para conjunciones dado el hecho de que se tratan como formulaciones atómicas independientes.

Fueron detectadas un conjunto de formulaciones que no son transformables dado que la notación lineal ORM2 no admite negaciones, estas incluyen la negación lógica, la formulación NAND, la formulación NOR y la formulación *si p ... o no q*.

No existe una formalización establecida para el caso de existencia de anillos, por ejemplo las reglas:

R1 Una persona no puede ser jefa de ella misma. (Anillo acíclico).

R2 Si persona x es colega de persona y, debe suceder también a la inversa. (Anillo simétrico).

R3 Ninguna persona es colega de ella misma. (Anillo irreflexivo).

Como se demostró anteriormente existen un conjunto de transformaciones para reglas de negocio expresadas en SBVR a la notación lineal ORM2 atendiendo al vocabulario del negocio y la formulación semántica de las reglas de negocio, pero estas transformaciones no pueden ser utilizadas por expertos del negocio ya que no existe una formalización de un algoritmo capaz de realizar este trabajo de forma automatizada para la detección de inconsistencias.

2.2. Validación y verificación de reglas de negocio

La verificación es el proceso dirigido a la detección de inconsistencia, Incompletitud o redundancia en un conjunto de reglas de negocio sin considerar el significado de las reglas, esto significa que no le importa acerca de si la regla es correcta o no (MEJIA, 2011).

Aun si se puede probar que las reglas son lógicamente consistentes y completas, las reglas podrían llevar a resultados incorrectos pero lo harán en una manera consistente.

La validación es el proceso que apunta por la detección de resultados incorrectos o comportamiento indeseado (MEJIA, 2011).

La forma más simple de validar reglas es pasar las reglas cambiadas a otro miembro de la organización. En un contexto de tecnologías de información, la validación frecuentemente se hace probando la aplicación y verificando los resultados, o comparando los resultados con previos resultados que se creen están correctos.

Algunos de los términos usados para señalar problemas en un conjunto de reglas se muestran a continuación (Castelo, 2011):

Inconsistencia: Es una condición en un conjunto de reglas de negocio que ocurre cuando 2 o más reglas llevan a comportamientos o resultados conflictivos.

Incompletitud: Es una condición en un conjunto de reglas que ocurre cuando existe un caso de negocio que lleva a un resultado o comportamiento indefinido.

Redundancia: Es una condición en un conjunto de reglas de negocio que ocurre cuando existe una regla de negocio que no tiene contribución significativa al posible comportamiento o posibles resultados.

Anomalía: Es un término más general para inconsistencia, incompletitud o redundancia.

Un conjunto de reglas posee alta calidad cuando no se producen las situaciones que se muestran a continuación (Ronald G. Ross, 2009):

Una regla es similar a otra regla.

Una regla subsume otra regla.

Una regla es lógicamente equivalente a otra regla.

Una regla está en conflicto con otra regla.

2.3. Algoritmo formal

El algoritmo que se propone constituye una guía en el proceso de transformación al formato ORM2 del conjunto de reglas de negocio que se analiza. La entrada al algoritmo es un conjunto de reglas de negocio representadas en el formato SBVR, como resultado se presenta la transformación al formato ORM2 de cada regla captada atendiendo a la estructura de la misma.

Entrada: La entrada del algoritmo son dos documentos en formato xml exportados desde SBVR. El primer documento contiene la terminología del negocio, o sea, el conjunto de entidades, variables y hechos que conforman el vocabulario del negocio. El segundo documento contiene el conjunto de reglas de negocio a ser analizado. Para cada regla describe su despliegue lógico permitiendo la captura de los elementos que conforman la regla.

Los elementos que conforman una regla constituyen operadores de la lógica proposicional, de la lógica de predicados y de la lógica modal, además de las entidades, variables o hechos que interactúan en la regla y que están previamente definidos en el documento que contiene el vocabulario del negocio.

Salida: Obtenido el conjunto de reglas transformadas desde SBVR a ORM2 se lleva a cabo el proceso de validación mediante la herramienta ORM2 to OWL2 Translator. Si solo existen anomalías depurables entonces se obtiene un conjunto de reglas consistentes. Si existe alguna anomalía que no sea posible eliminar entonces se señala entre que pares de reglas se presenta. El conocedor del negocio decide cómo tratar dicha inconsistencia.

1. Captar el conjunto de entidades, variables y hechos descritos en el documento que contiene el vocabulario del negocio.

- 1.1. Para cada entidad y variable declararla en formato ORM2.

- 1.2. Para cada hecho descrito en el documento de vocabulario de negocio, realizar la transformación equivalente a ORM2.
2. Captar el documento que contiene las reglas.
 - 2.1. Leer reglas de negocio expresadas en SBVR.
 - 2.1.1. Para cada regla de negocio expresada en SBVR.
 - 2.1.2. Si la regla posee un cuantificador universal y este actúa sobre entidades, el primer operando lógico de la disyunción es una formulación de instanciación por lo que la transformación equivalente de la regla a ORM2 es: **De disyunción con respecto a entidades.**
 - 2.1.3. Si la regla posee un cuantificador universal y este actúa sobre entidades donde existe una relación entre éstas, el primer y segundo operando lógico de la disyunción es una formulación atómica por lo que la transformación equivalente de la regla a ORM2 es: **De disyunción con respecto a relaciones.**
 - 2.1.4. Si la regla posee una cuantificación universal donde intervienen entidades y este se extiende sobre una cuantificación existencial, la cuantificación existencial se extiende sobre una formulación atómica. La transformación equivalente a ORM2 es: **Roles de mandamiento.**
 - 2.1.5. Si la regla posee una cuantificación como-máximo-uno y esta introduce una variable, la cuantificación como-máximo-uno se extiende sobre una formulación atómica donde la formulación atómica es basada en el tipo de hecho que interviene en la regla. La transformación equivalente a ORM2 es: **Restricción de frecuencia.**
 - 2.1.6 Si la regla posee una cuantificación universal que actúa sobre alguna entidad y ésta cuantificación universal se extiende sobre una cuantificación al-menos-n. La transformación equivalente a ORM2 es: **Restricción de frecuencia con cuantificación al-menos-n.**
 - 2.1.7 Si la regla posee una cuantificación al-menos-n con $N \geq 2$ donde existe alguna variable y la misma está restringida por una objetificación. La transformación equivalente a ORM2 es: **Variable varía sobre fact type restringida por una objetificación.**
 - 2.1.8 Si en la regla existe una formulación atómica basada en un tipo de hecho donde el antecedente de la implicación es un cuantificador existencial, el cuantificador existencial

introduce una variable y se extiende sobre una formulación atómica. La transformación equivalente a ORM2 es de: **Implicación**.

2.1.9 Si en la regla existe una objetificación, donde la misma enlaza una variable, la objetificación considera una formulación atómica donde la formulación atómica es basada en un tipo de hecho. La transformación equivalente a ORM2 es: **Variable restringida por una objetificación**.

2.1.10 Si la regla posee una variable donde la variable esté relacionada con una proposición de nominalización, la proposición de nominalización enlaza a la variable y existe una formulación atómica basada en un tipo de hecho. La transformación equivalente a ORM2 es: **Variable restringida por una proposición de nominalización**.

3 Detectar posibles contradicciones en el conjunto de reglas de negocio.

3.1. Mientras se detecte alguna contradicción:

3.1.1. Se especifica entre que reglas se presenta. El especialista o conocedor del negocio, es el encargado de realizar la depuración.

4 .Eliminar posibles redundancias en el conjunto de reglas de negocio.

5 .Obtención de un documento de reglas consistentes.

Los pasos tres y cuatro y cinco se explican a continuación:

Paso 3: Este apunta a la detección de contradicciones. Se analiza el conjunto de reglas en formato ORM2. Si se está en presencia de dos reglas que conducen a situaciones contradictorias al menos una de las reglas pudiera estar mal declarada, el experto del negocio es quien debe decidir cómo tratar esta contradicción.

Paso 4: Se dedica a la eliminación de redundancias, en este paso es tomado el conjunto de reglas en formato ORM2 por la herramienta ORM2 to OWL Translator, al ejecutar la herramienta, si existen redundancias se especifica entre que pares de reglas se presenta, es responsabilidad del programador eliminar la regla redundante. Este paso se realiza hasta que no existan redundancias.

Paso 5: se dedica a la obtención de un documento con el conjunto de reglas consistente, aquí se realiza un mapeo de las reglas obtenidas en el formato OWL a las reglas originales descritas en SBVR definidas en el lenguaje del usuario del negocio y de la herramienta de captura y edición de reglas de negocio.

Problemas que presenta el algoritmo:

El algoritmo de transformación no puede transformar todas las reglas de negocio expresadas en SBVR al formato ORM2, ya que la notación lineal ORM2 no admite negaciones por lo que constituye un problema al no poder realizar su validación.

2.4. Conclusiones parciales del capítulo

En este capítulo se formalizó un algoritmo para transformar reglas de negocio expresadas sobre SBVR hacia el formato ORM2 a partir del basamento de un conjunto de formalizaciones semánticas del SBVR, así como su propuesta de transformación hacia la notación lineal ORM2 soportada sobre una semántica FOL(Lógica de primer orden). Se hace énfasis en la validación y verificación de reglas de negocio donde se tratan algunos de los términos usados para señalar problemas a ocurrir en un conjunto de reglas.

CAPÍTULO 3.

EJEMPLO DE APLICACIÓN DEL ALGORITMO PARA EL CASO DE ESTUDIO: CONTROL DE PACIENTES EN EL ÁREA DE NEFROLOGÍA.

CAPÍTULO 3. EJEMPLO DE APLICACIÓN DEL ALGORITMO PARA EL CASO DE ESTUDIO: CONTROL DE PACIENTES EN EL ÁREA DE NEFROLOGÍA.

Los servicios de salud se caracterizan por un control de la evolución de los pacientes en consultas generales y, en la medida que los diagnósticos ganan en precisión y los casos así lo requieren, su continuidad en consultas especializadas. Esta práctica médica, aparentemente sencilla, supone un respaldo informativo complejo porque según cómo se comportan los casos, requieren de tratamiento diferenciado donde difícilmente se pueden abordar soluciones generales. Desde el punto de vista computacional la coordinación y realización de las actividades requeridas, usualmente, implican el uso de tecnologías que responden a diferentes paradigmas y, sobre esta base, las aplicaciones pueden variar en dependencia del peso que se le dé a alguno de ellos. Uno de los objetivos de esta investigación es la aplicación de la formalización del algoritmo para un ejemplo específico relacionado con el caso de estudio: control de pacientes en el área de nefrología. En este capítulo se desarrolla mediante un ejemplo la aplicación de este algoritmo, así como algunas de las inconsistencias que pueden ocurrir en las reglas de negocio.

3.1. Caso de estudio: control de pacientes en el área de nefrología

El área médica, en especial nefrología, se caracteriza por reglamentaciones especificadas en complejos protocolos que son frecuentemente modificados y varían con relación a criterios científicos, por países, regiones, localidades e incluso atendiendo a consideraciones religiosas. En este epígrafe se muestra el área de nefrología como caso de estudio para aplicar el enfoque de reglas de negocio (M. E. Martínez Busto, Mendilahaxón Valdés, Cuellar Vega, & González González 2009).

Los procedimientos terapéuticos en el área de la salud se rigen por complejos protocolos de trabajo. En el área de nefrología, como caso particular, un paciente puede pasar por diferentes tipos de consultas según sean sus avances y evolución. Médicos y paramédicos deben tomar importantes decisiones ante cada nueva situación acorde a los resultados en los exámenes realizados y siguiendo los protocolos que rigen todo el proceso. Los pacientes que son tratados en esta área tienen, desde hace pocos años, como primera opción terapéutica el trasplante de riñón.

Con los avances científicos y los resultados investigativos en este campo de la ciencia, las regulaciones para el seguimiento de los pacientes en dicha área suelen ser dinámicas, por lo que,

los SI que necesariamente deben apoyar estos procesos deben también adaptarse a dichos cambios.

3.1.1. Control de los pacientes en el área de nefrología

El estado de un paciente se puede determinar en base a diferentes parámetros, algunos son propios del área de nefrología y otros son de interés para otras áreas o especialidades, pero todos deben ser adecuadamente monitoreados pues garantizan la atención integral al paciente.

Esto sugiere manejar al paciente como entidad abstracta, accesible desde cualquier consulta, con la seguridad de disponer de una vista actualizada del mismo y a la vez, ofrecer vistas para cada consulta en particular. Se debe considerar que el paso de un paciente por alguna de estas consultas generalmente obedece a rigurosas regulaciones.

Los pacientes que se atienden en el área de nefrología reciben atención especializada, insertándose en un sistema de atención secundaria y terciaria personalizada. Para llevar a cabo dicha atención se requiere registrar y gestionar de manera permanente los exámenes y la evolución para cada paciente. El área de nefrología incluye diversas especialidades que, en dependencia de cada uno estos controles, tiene sus peculiaridades. La atención al paciente en esta área se realiza de acuerdo al seguimiento en tres tipos de consultas: *consulta de progresión o renal crónica avanzada (externa o evolutiva)*; *consulta de hemodiálisis* y *consulta de trasplante*.

Cada consulta tiene sus propios requisitos de información, protocolos de trabajo y restricciones, que deben ser considerados tanto en la práctica médica como para determinar requisitos y funcionalidades esperados en los sistemas computacionales de apoyo. Esto resulta de gran importancia, no sólo para el personal médico y paramédico sino para pacientes y familiares relacionados con esta área. Esto posibilita que no sólo se conozca acerca del estado actualizado del enfermo, sino que adquieran cultura en cuanto a los cuidados que se deben tener con pacientes con estas características, donde la atención familiar resulta decisiva en el éxito del tratamiento.

Cuando un paciente comienza a tratarse dentro del área de nefrología es por haber encontrado parámetros que indican la posibilidad de padecer una insuficiencia total o casi total en el funcionamiento del riñón, enfermedad conocida como insuficiencia renal crónica (IRC). Cuando esta enfermedad está en una etapa terminal, desenlace común a múltiples enfermedades que

afectan al riñón, se denomina insuficiencia renal crónica terminal (IRCT). Esta se presenta cuando los riñones ya no pueden funcionar al nivel necesario para la vida diaria, es decir, que la IRC progresa a un punto tal que la función de los riñones es menor del 10% de su capacidad normal. El área de nefrología atiende pacientes en los diferentes estadios de esta enfermedad.

Determinar el estadio en que se encuentra cada paciente en base a los resultados de los exámenes que se orientan en cada consulta es una tarea fundamental dentro del área y de acuerdo a ello será el tratamiento que reciba.

Todo paciente con diagnóstico de *IRCT*, con independencia del área en que se hizo el diagnóstico y de preferencia en estadio pre dialítico o una vez que esté siendo atendido con un método depurador, debe ser siempre valorado en una consulta de nefrología. Allí se explica al paciente las opciones de tratamiento, incluido el *TR*.

Los posibles estadios de un paciente se clasifican del I al V. De encontrarse el paciente en alguno de los dos primeros es reorientado a su área de salud para ser atendido allí en la consulta de nefrología. Si se encuentra en uno de los *estadios III* o *IV* el paciente es atendido en la propia consulta de progresión. Cuando el paciente se encuentra en *estadio V* se considera que la enfermedad se encuentra en estado terminal y el paciente es remitido a uno de los métodos sustitutivos de la función renal: *diálisis peritoneal, hemodiálisis o trasplante renal*.

La terapia con *diálisis peritoneal* puede representar una opción para salvar la vida de aquellos pacientes con IRCT que no pueden, por alguna razón, recibir tratamiento de trasplante del órgano.

El órgano para el trasplante puede provenir de dos fuentes: (1) de un donante familiar vivo, genéticamente emparentado con el receptor, padres, hermanos o hijos (Kasiske et al., 1995) y (2) de un donante muerto o donante con muerte encefálica (persona recientemente fallecida que no ha tenido enfermedad renal crónica)(A. Ojo et al., 2001; A. Ojo, Hanson, J., HU., M.K., Okechukwu, C., Wolfe, R., Liechtman, A., Port., F, 2001) Para conformar la *pareja receptor-donante potencial* se establecen dos tipos posibles de donantes de acuerdo el origen del órgano, considerando que es posible tener uno o varios donantes potenciales y uno o varios posibles receptores. La pareja constituida es sometida a un protocolo de compatibilidad.

En caso de tener un resultado favorable, al finalizar el *protocolo de compatibilidad*, se envían los pacientes a la consulta de TR o *consulta multidisciplinaria*. O sea, el receptor y posibles donadores o donante potencial asociados (se pueden proponer uno o varios donadores candidatos) son remitidos a esta consulta, integrada de varios especialistas, entre ellos: *nefrólogo, cirujano y urólogo*, y otros; generalmente es dirigida por un nefrólogo (Hernández, 2003), donde se evalúa la pareja *receptor-donador potencial*.

El *donante* juega un papel muy importante dentro de este proceso. Identificar el mejor entre los posibles donantes es una delicada tarea guiada por un complejo protocolo. Cada uno de los exámenes va descartando candidatos para, finalmente, elegir la mejor posibilidad de éxito en la operación. Una vez completado el *protocolo para donante potencial*, si no se rechazan todos, se pasa a seleccionar el *donante*. En caso exitoso se conforma la *pareja donante-receptor* y se da continuidad al *protocolo de trasplante*.

La *operación de trasplante* asociada a un *donante con muerte encefálica* sigue un procedimiento muy similar a la correspondiente a *donante vivo*, difieren básicamente respecto a la operación de extracción del órgano del donante y al tratamiento del donador. Los registros respecto al donante también difieren, pero en cuanto a otras actividades básicas son exactamente iguales: los protocolos y registros para el órgano donado, la operación de extracción e implante del mismo, y el *seguimiento postoperatorio* para el receptor.

Con el fin de evitar el rechazo casi todos los receptores de trasplante de riñón requieren tratamiento de por vida. El éxito de un trasplante de riñón depende, en parte, del seguimiento minucioso y el cumplimiento meticuloso del régimen de medicamentos. Los pacientes deben ser chequeados periódicamente, controlando un conjunto de parámetros que evidencian posibles padecimientos o complicaciones. El seguimiento al que es sometido el paciente posterior a la operación depende de su desarrollo y evolución. Esta consulta es llamada *consulta de progresión, externa o evolutiva*.

Las únicas posibles formas en que un enfermo puede salir del sistema son por la muerte del paciente o debido a que cualquier paciente o posible donante abandone de forma voluntaria el tratamiento.

3.1.2. Caracterización del caso de estudio

De la descripción anterior del caso de estudio, ampliada en (M. E. Martínez Busto et al., 2009), se concluye que:

1. Durante la atención a un paciente en esta área se tienen que tomar múltiples decisiones que están muy bien reglamentadas en diferentes protocolos.
2. Las decisiones están fuertemente interrelacionadas, o sea, unas pueden depender de otras.
3. Cada usuario, médico, paramédico, paciente o familiar tiene necesidades informativas diferentes. Debe propiciarse el acceso y manejo de la información a cada cual según sus intereses.
4. Los tratamientos y procedimientos son susceptibles a cambios, e incluso hasta pueden variar de acuerdo a la región o país, se rigen por normativas y pueden variar de acuerdo a resultados experimentales y a los avances científicos en el área.

Las regulaciones médicas son altamente sensibles y deben ser solamente establecidas por personal altamente capacitado y responsabilizado con una especialidad dada, en este caso, la especialidad de nefrología.

Todo lo anterior sugiere que aplicar el *enfoque de reglas de negocio* facilita la obtención de soluciones computacionales para este dominio. El sector de la salud es un área en que los cambios son muy frecuentes dado el vertiginoso desarrollo que tiene la medicina, tanto en nuestro país como a nivel mundial. Estos cambios se originan en la comunidad científica y se canalizan a través del sistema de salud hasta su instrumentación por parte de los especialistas, que incluso pueden aportar, con su experiencia, determinadas variaciones. Lo anterior, refuerza el hecho que deben ser los propios especialistas del negocio, en este caso los médicos, los encargados de crear y dar mantenimiento a sus políticas, expresadas con una terminología que depende en gran medida de cada área o sector. Hay que tener en cuenta las reglamentaciones y normativas que rigen o restringen los diferentes procesos y actividades en un negocio, no sólo en la etapa inicial del ciclo de vida de un SI, sino también a lo largo del resto de las diferentes etapas, especialmente en sectores tan sensibles como el de la salud.

Un SI que complemente los servicios médicos, debe ser coherente con las observaciones anteriores, y para ello debe:

- Recopilar información en la etapa de análisis de los requisitos del software lo que permite crear y mantener los registros relacionados con la actividad y manejo de pacientes del área de nefrología y su seguimiento.
- Determinar cuáles reglamentaciones rigen esta área, según los protocolos de trabajo establecidos.
- Determinar el vocabulario médico utilizado por la comunidad profesional ligada a cada etapa o actividad concreta.

Es obvio que los SI requieren de un cuidadoso diseño, algo que en el sector de la salud es particularmente importante debido a lo sensible del área y la aplicación de tecnologías que responden a diferentes paradigmas computacionales. En la actualidad el *enfoque de reglas de negocio* coloca en un primer plano la captación de las políticas, regulaciones, leyes, que deben hacerse cumplir o simplemente observar durante los procesos que se llevan a cabo.

3.2. Aplicación del algoritmo a una muestra del caso de estudio.

Ejemplo:

En un trabajo de diploma del año anterior titulado Formalización y validación de consistencia para reglas de negocio expresadas sobre SBVR (Parra, 2013) para el chequeo de la consistencia de las reglas expresadas en notación lineal ORM2 se utilizó la herramienta “ORM2 to OWL2 Translator”.

Esta herramienta para la validación de conflictos e inconsistencias usando las reglas transformadas a la notación lineal ORM2, fue aplicada al caso de estudio de trasplante renal. Se analizaron con la herramienta un conjunto de reglas de negocio extraídas del caso de estudio del área de nefrología, con el objetivo de identificar posibles contradicciones entre dichas reglas.

De ellas se tomó una muestra para el desarrollo del caso de estudio, a continuación se muestran las reglas seleccionadas:

R3 Todo donante potencial es un donante.

R7 Todo donante candidato es un donante.

R11 Todo donante vivo o cadavérico es un donante.

R15 Todo examen realizado es un examen.

R18 Todo donante es un paciente.

R22 Todo receptor es un paciente.

- R26 Todo examen físico es un examen.
- R29 Todo examen complementario es un examen.
- R33 Todo trasplante renal tiene que tener asociado un donante vivo o donante cadavérico.
- R36 Todo donante potencial puede estar asociado a un receptor.
- R40 Todo antecedente patológico tiene asociado un paciente.
- R43 Todo donante potencial tiene asociado un paciente.
- R47 Todo examen físico está asociado a un paciente.
- R50 Un paciente puede ser receptor.
- R53 Todo antecedente patológico tiene asociado al menos una enfermedad.
- R57 Todo examen realizado tiene que ser asociado a un paciente.
- R60 Los exámenes asociados al protocolo de trasplante tienen que ser válidos durante el trasplante renal.
- R63 El resultado de los análisis de los pacientes es desechado después del trasplante renal.
- R66 Todo examen realizado al receptor durante el protocolo de trasplante tiene que ser almacenado asociado a protocolo de trasplante.
- R69 Todo examen realizado al donante durante el protocolo de compatibilidad tiene que ser almacenado asociado a protocolo de compatibilidad.
- R73 Se necesita valorar los resultados de análisis complementarios del paciente del protocolo de trasplante para chequear sus condiciones físicas.
- R76 Solo cuando las condiciones físicas del pacientes son favorables se realiza análisis de características por sexo.
- R79 Solo cuando las condiciones mentales de los pacientes son favorables se realiza análisis de características por sexo.
- R82 Solo cuando el examen complementario de VIH del paciente fue negativo se realiza el trasplante renal.
- R85 Solo cuando el examen complementario de VIH del donante fue negativo se realiza el trasplante renal.
- R88 Después de concluir el protocolo de compatibilidad exitosamente se tiene un conjunto de donantes candidatos.
- R91 La frecuencia con la que se realiza el examen de hemoglobina en la consulta de hemodiálisis es mensual.
- R94 La operación de trasplante se realiza en un intervalo de 2 a 4 horas.
- R97 La operación de trasplante tiene que ser desarrollado solo cuando el protocolo de compatibilidad se haya realizado.

A partir de las mismas se seleccionan algunas para aplicar el algoritmo de transformación de reglas de negocio expresadas en SBVR al formato ORM2.

A continuación se presenta la siguiente regla de negocio número 11:

Todo donante vivo o cadavérico es un donante

Teniendo en cuenta los elementos que conforman la terminología del vocabulario del negocio (el conjunto de entidades, variables, operadores de la lógica proposicional, de la lógica de predicados y de la lógica modal, además de hechos que interactúan en la regla).

- La regla anterior está compuesta por las entidades: **donante vivo, donante cadavérico y donante.**
- Un cuantificador Universal: **Todo.**
- Un tipo de hecho: **Los donantes vivos o cadavéricos son donantes.**

Cuando aplicamos el algoritmo de acuerdo a la entrada del mismo y una vez leído la regla de negocio expresada en SBVR se le aplica el paso 2.1.2 Por lo que la transformación de la regla atendiendo a la notación lineal ORM2 quedaría de la siguiente forma:

Tipo de transformación: Disyunción con respecto a entidades.

Sintaxis: $O-SETexc(\{Obj_2, Obj_3\}, Obj_1)$

Donde los Obj representan los <fact type role> es decir objetos basados en su rol, asumiendo una función o siendo usado en alguna situación según sus índices.

Sustitución:

$O-SETex(\{Donante_Vivo, Donante_Cadaverico\}, Donante)$

A continuación se presenta la siguiente regla de negocio número 33:

Todo trasplante renal tiene que tener asociado un donante vivo o donante cadavérico

- La regla anterior está compuesta por las entidades: **Trasplante renal, donante vivo, donante cadavérico.**
- Un cuantificador Universal: **Todo**
- Un tipo de hecho: **Los trasplantes renales tiene que tener donantes vivos o cadavéricos.**

Cuando aplicamos el algoritmo una vez leído la regla de negocio expresada en SBVR en este caso se le aplica el paso 2.1.3 Por lo que la transformación de la regla atendiendo a la notación lineal ORM2 quedaría de la siguiente forma:

Tipo de transformación: Disyunción con respecto a relaciones.

Sintaxis: *LOC-ROLES-INDEX*(*Rel*₁*#. var*_{card1}, 1)

TYPE(*Rel*₁*#. var*_{card1}, *UppercaseWord*(*var*_{card1}))

LOC-ROLES-INDEX(*Rel*₁*#. var*_{card2}, 2)

TYPE(*Rel*₁*#. var*_{card2}, *UppercaseWord*(*var*_{card2}))

LOC-ROLES-INDEX(*Rel*₂*#. var*_{card1}, 1)

TYPE(*Rel*₂*#. var*_{card1}, *UppercaseWord*(*var*_{card1}))

LOC-ROLES-INDEX(*Rel*₂*#. var*_{card2}, 2)

TYPE(*Rel*₂*#. var*_{card2}, *UppercaseWord*(*var*_{card2}))

*R-SET*_{exc}({*Rel*₁*#. var*_{card1}}, {*Rel*₂*#. var*_{card1}}, {*Rel*₁*#. var*_{card1}, *Rel*₂*#. var*_{card1}})

Sustitución:

LOC-ROLES-INDEX(*tiene_asociado33.trasplante_renal*, 1)

LOC-ROLES-INDEX(*tiene_asociado33.donante_vivo*, 2)

TYPE(*tiene_asociado33.trasplante_renal*, *Trasplante_Renal*)

TYPE(*tiene_asociado33.donante_vivo*, *Donante_Vivo*)

LOC-ROLES-INDEX(*tiene_asociado33_1.trasplante_renal*, 1)

LOC-ROLES-INDEX(*tiene_asociado33_1.donante_cadaverico*, 2)

TYPE(*tiene_asociado33_1.trasplante_renal*, *Trasplante_Renal*)

TYPE(*tiene_asociado33_1.donante_cadaverico*, *Donante_Cadaverico*)

*R-SET*_{exc}({*tiene_asociado33.donante_vivo*},

{*tiene_asociado33_1.donante_cadaverico*},

{{*tiene_asociado33.donante_vivo*, *tiene_asociado33_1.donante_cadaverico*}})

A continuación se presenta la siguiente regla de negocio numero 53:

Todo antecedente patológico tiene asociado al menos una enfermedad.

- La regla anterior está compuesta por las entidades:
 1. **Antecedente patológico.**
 2. **Enfermedad.**
- Un cuantificador Universal: **Todo.**
- Una cuantificación **al-menos-n.**

Una vez aplicado el algoritmo la transformación de la regla atendiendo a la notación lineal ORM2 sería de la forma siguiente:

Tipo de transformación: **Restricción de frecuencia con cuantificación al-menos-n.**

Sintaxis: $R-CARD(Rel\#) = (n, INF)$

Sustitución:

$LOC-ROLES-INDEX(tiene_asociado53. antecedente_patologico, 1)$

$LOC-ROLES-INDEX(tiene_asociado53. enfermedad, 2)$

$TYPE(tiene_asociado53. antecedente_patologico, Antecedente_Patologico)$

$TYPE(tiene_asociado53. enfermedad, Enfermedad)$

$R-CARD(tiene_asociado53) = (1, INF)$

Ejemplo de inconsistencias:

Ejemplo#1: Reglas en las que pueden ocurrir inconsistencias.

R1 Todo donante vivo es un donante.

R2 Todo donante cadavérico es un donante.

R3 Todo donante vivo o cadavérico es un donante.

En este caso estamos en presencia de reglas redundantes donde la regla 3 cubre todos los casos de las reglas 1 y 2.

Solución:

En este caso se eliminan las reglas 1 y 2 para solo dejar la regla más general.

Ejemplo#2:

R1 La frecuencia con la que se realiza el examen de hemoglobina en la consulta de hemodiálisis es mensual.

R2 La frecuencia con la que se realiza el examen de hemoglobina en la consulta de hemodiálisis es semanal.

En este caso estamos en presencia de una contradicción ya que las regla 1 y la regla 2 son dependientes pero presentan acciones diferentes por tanto al menos una de las reglas está mal esbozada.

Solución:

En este caso el experto del negocio es quien debe decidir cómo tratar esta contradicción.

Ejemplo#3:

R 79 Solo cuando las condiciones mentales del paciente son favorables se realiza análisis de características por sexo.

R 76 Solo cuando las condiciones físicas del paciente son favorables se realiza análisis de características por sexo.

En este caso estamos en presencia de una contradicción ya que las regla 79 y la regla 76 son dependientes pero presentan acciones diferentes por tanto al menos una de las reglas está mal esbozada.

Solución:

En este caso el experto del negocio es quien debe decidir cómo tratar esta contradicción.

3.4 Conclusiones parciales del capítulo

En este capítulo se realiza la aplicación del algoritmo de transformación de reglas de negocio basado en un conjunto de formalizaciones semánticas del SBVR y su transformación hacia la notación lineal ORM2. Se aplica el algoritmo para una muestra del caso de estudio, ilustrando mediante ejemplos como se realiza la transformación de reglas de negocio hacia el formato ORM2. Se hace énfasis mediante ejemplos sobre la detección de contradicciones y redundancias, así como el tratamiento que deben tener en cada caso estos tipos de inconsistencias producto del análisis de las reglas de negocio expresadas sobre SBVR.

Conclusiones

Como resultado de esta investigación se concluyó que basados en la formalización de un algoritmo es posible la transformación de reglas de negocio expresadas sobre SBVR hacia el formato ORM2 para realizar validaciones semánticas con el objetivo de eliminar inconsistencias en el repositorio de reglas de negocio.

1. Se realizó un estudio sobre las formalizaciones propuestas por el SBVR.
2. Se propuso la formalización de un algoritmo para transformar reglas de negocio expresadas sobre SBVR hacia el formato ORM2 basados en el formato de intercambio de SBVR hacia la notación lineal, y se describen los pasos para las diferentes situaciones.
3. Se aplica la formalización del algoritmo propuesto mediante varios ejemplos de reglas de negocio para una muestra del caso de estudio del área de nefrología.

Recomendaciones

1. Implementar el algoritmo de transformación propuesto y su incorporación a la herramienta UCRules para la administración de reglas de negocio.
2. Dar continuidad al estudio mediante la incorporación de otros métodos para el análisis de la consistencia del conjunto de reglas de negocio.
3. Estudiar la nueva versión del SBVR en cuanto a posibles actualizaciones y formalizaciones para las restricciones de anillos.

Referencias bibliográficas

- Bajec, M., Krisper, M., & Rupnik, R. (2000). *Using Business Rules Technologies To Bridge The Gap Between Business and Business Applications*. Paper presented at the The IFIP 16th World Computer Congress 2000, Information Technology for Business Management, Peking, China.
- Bollen. (2009). The orchestration of fact-orientation and SBVR. *Business Information Processing., (Pubs) Springer*, 302–312pp.
- BRG. (2003). *The Business Rule Manifesto - The Principles of Rule Independence* (Vol. 2008).
- Castelo, C. A. M. (2011). *Investigación para la adopción y uso de los motores de reglas de negocio*. Licenciado, San Buena Ventura, Santiago de Cali.
- Franconi, E., Mosca, A. & Solomakhin, D. (2011). ORM2: Syntax and semantics. Internal report. KRDB Research Centre for Knowledge and Data.
- Guedes, D. (2013). *Administración de reglas de negocio desde la perspectiva del negocio basado en SBVR*. Licenciado, Universidad Central de Las Villas, Santa Clara.
- Halpin, T. M., T. (1989). A Logical Analysis of Information Systems: Static Aspects of the Dataoriented Perspective. *Department of Computer Science, University of Queensland, Australia*. 222p.
- Halpin, T. M., T. (2008). Information Modeling and Relational Databases. *Morgan Kaufmann. San Francisco, CA, USA*. 976 pp.
- Hay, D. C., Kolber, A., & Anderson, K. H. (1997). GUIDE Business Rules Project: Final Report *GUIDE International Corporation*.
- Hernández, P. E., López, J., Cruz, R. E., Pérez, R. A., López, J., Batista, R. & Hernández, O. (2003). *Protocolo de Trasplante Renal Donador Vivo Relacionado*. Hospital Universitario “Arnaldo Milián Castro” MINSAP Santa Clara.
- Kasiske, B., Cangro, C., Harihanan, S., Hricik, D., Kerman, R., Roth, D., . . . Weir, M. (1995). The evaluation of renal transplant candidates: Clinical practice guidelines. *J Am Transplant 2(suppl 1)*, 5-95.
- Marrero, V. (2013). *Tablas de decisión para el análisis de consistencia y chequeos de integridad de un conjunto de reglas de negocio expresadas en SBVR*. Licenciado, Universidad Central de Las Villas, Santa Clara.
- Martínez Busto, M. E., Caballero Martínez, A., Guedes Coello, D., nuñez, C. A. & González González , L. (2012). *Editor de reglas “UCRules*. Universidad Central "Marta Abreu" de Las Villas, UCVL Santa Clara.Cuba.
- Martínez Busto, M. E., Mendilahaxon Valdés, L. F., Cuellar Vega, E., & González González , L. (2009, June 2-5, 2009). *Modelación de reglas y procesos para la prescripción medica y el control de medicamentos en farmacia intrahospitalaria*. Paper presented at the Seventh LACCEI Latin American and Caribbean Conference for Engineering and Technology (LACCEI'2009), “Energy and Technology for the Americas: Education, Innovation, Technology and Practice”, San Cristóbal, Venezuela.
- Aplicación de las reglas de negocio a sistemas inter-hospitalarios para trasplante renal en Cuba (2009).

- MEJIA, C. A. (2011). *Investigación para la adopción y uso de los motores de reglas de negocio.*, Universidad San Buenaventura, Santiago de Cali.
- Morgan, T. (2002). *Business Rules and Information Systems: Aligning IT with Business Goals:* Addison Wesley.
- Núñez, L. (2012). *Validación de la consistencia de un conjunto de reglas de negocio.* Maestría, Universidad Central de Las Villas, Santa Clara.
- Ojo, A., Hanson, J., HU., M.-K., Okechukwu, C., Wolfe, R., Liechtman, A., . . . Port, F. (2001). Survival in recipients of marginal cadaveric donor kidneys, compared to other recipients and wait-listed transplant patients. *J Am Soc Nephrol* 12, 589-597.
- Ojo, A., Hanson, J., HU., M.K., Okechukwu, C., Wolfe, R., Liechtman, A., Port., F. (2001). Survival in recipients of marginal cadaveric donor kidneys, compared to other recipients and wait-listed transplant patients. *J Am Soc Nephrol*, 12, 589-597.
- OMG. (2008). *Semantics of Business Vocabulary and Business Rules (SBVR), v1.0.*
- Parra, J. C. (2013). *Formalización y validación de consistencia para reglas de negocio expresadas sobre SBVR.* Licenciado, Universidad Central de Las Villas, Santa Clara.
- Ross, R. G. (2000). What are Fact Models and why do you need them? . *BRCommunity.com*. Retrieved from <http://www.brcommunity.com/cgi-bin/x.pl/commentary/b008a.html>, <http://www.brcommunity.com/cgi-bin/x.pl/commentary/b008b.html>
- Ross, R. G. (2003). *Principles of the Business Rule Approach:* Addison-Wesley Professional.
- Ross, R. G. (2009). *Business Rule Concepts* (3 ed.).
- Solomakhin, D. (2011). On the logical foundations of business rules. *Organisational Systems and Strategies*.
- Weiden, M., Hermans, L., Schreiber, G., & van der Zee, S. (2004). Classification and Representation of Business Rules. *Última visita: Enero del 2013*. Retrieved from www.omg.org/docs/ad/02-12-18.pdf
- Youdeowei, A. (1997). *The B-Rule Methodology: A Business Rule Approach to Information Systems Development.* doctorado, Manchester, United Kingdom.