

# **Universidad Central "Marta Abreu" de Las Villas.**

**Facultad Matemática – Física – Computación**



*"Validación semántica de términos médicos en UMLS."*

*Proyecto Trabajo de Diploma.*

**Autor:** Michael Pérez Rodríguez

**Tutores:** MSc. Manuel Castro

Dra. Luisa González.

**Villa Clara, Cuba.  
2008**

## **Pensamiento**

“No nos falta valor para emprender ciertas cosas porque son difíciles,  
sino que son difíciles porque nos falta el valor para emprenderlas.”

Séneca.

## ***Agradecimientos***

A mi familia por estar siempre que lo necesitaba.

A mis compañeros de estudio José Rafael y Lisandra que aguantaron mis locuras.

Agradezco especialmente a:

Lic. Isel Moreno.

Msc. Maria Elena Martínez.

Msc. Manuel Castro.

Dra. Luisa Gonzáles.

...y a todos los que en general brindaron su ayuda en la realización de este proyecto.

## ***Resumen***

El presente trabajo consiste en la creación de un sistema cuyo fin sea la validación semántica de Reglas de Negocio relacionadas con los procesos médicos, específicamente con los procesos de nefrología. El analizador semántico se implementa basado en el *Unified Medical Language System (UMLS)* utilizando las clases y funciones que brinda el *UMLS* para la búsqueda y consulta de términos médicos especializados contenidos en una ontología. A partir de una aplicación contenida en el *UMLS* con estilo de navegador, se implementa en lenguaje Java el analizador semántico garantizando la validación semántica de las Reglas de Negocio.

## **Abstract**

The present work consists on the creation of a system whose end is the semantic validation of Business Rules related with the medical processes, specifically with the nefrology processes. The semantic analyzer is implemented based on the Unified Medical Language System (UMLS) using the classes and functions that the UMLS offer for the search and consultation of terms medical specialized contents in an ontology. Starting from an application contained in the UMLS with navigator style, it is implemented in language Java the semantic analyzer guaranteeing the semantic validation of the Business Rules.

## ***Índice***

Resumen .....	4
Índice .....	6
Introducción.....	7
Capítulo 1 .....	10
Conceptos y herramientas usadas para la realización del Sistema para la Validación Semántica de Reglas de Negocio.....	11
1.1 Uso de las Ontologías en las ramas de investigación: .....	11
1.1.1 Concepto de ontología. ....	11
1.1.2 Tipos de Ontologías.....	12
1.1.3 Otras definiciones. ....	13
1.2 Enfoque de Reglas de Negocio.....	13
1.2.1. Patrones para Reglas de Negocio .....	14
1.2.2 Forma moderna de implementación de Reglas de Negocio.....	17
1.3 Sistema de Lenguaje Medico Unificado (UMLS).....	18
1.3.1 Caracterización del Sistema de Lenguaje Médico Unificado. ....	20
1.3.2 Metathesaurus.....	20
1.3.3 Red Semántica. ....	24
1.3.4 SPECIALIST Lexicon.....	28
Capítulo 2 .....	31
Diseño e implementación del Sistema para la Validación Semántica de Reglas de Negocio.....	32
2.1 Diseño del Sistema. ....	32
2.1.1 Arquitectura del editor de Reglas de Negocio. Localización del módulo para la validación semántica de Reglas de Negocio. ....	32
2.1.2 Análisis y diseño del analizador semántico. ....	33
2.1.3 Diagrama de Actividades.....	33
2.1.4 Actores del Sistema y sus funciones.....	33
2.2 Casos de Uso. ....	34
2.2.1 Descripción de los Casos de Uso.....	34
2.3 Diseño de la interfaz del sistema. ....	36
2.3.1 Herramientas utilizadas para el desarrollo del diseño.....	37
2.4 Implementación del módulo de validación de Reglas de Negocio. ....	37
2.4.1 Herramienta para el manejo de los términos especializados.....	38
2.4.2 Herramienta para la búsqueda de los términos especializados y la validación de reglas de negocio. ....	38
2.4.3 Función de validación desde un fichero externo.....	42
Capítulo 3 .....	45
Guía del Sistema de Validación Semántica de Reglas de Negocio. ....	46
3.1 Características generales del Sistema. ....	46
3.1.1 Especificaciones del MetamorphoSys. ....	47
3.1.2 Instalación. ....	47
3.2 Guía del Analizador Semántico. ....	52
3.2.1 Validación en el explorador.....	54
3.2.2 Validación desde fichero externo. ....	57
3.2.3 Consulta de términos independientes. ....	59
Conclusiones.....	60
Recomendaciones .....	62
Referencias Bibliográficas.....	64
Anexos.....	65

## **Introducción**

El desarrollo de la computación en los últimos años ha permitido un incremento del uso de las herramientas informáticas por parte de una comunidad de usuarios que crece aceleradamente. Esto implica que la tendencia actual sea automatizar la mayor cantidad de procesos complejos que se presentan en la vida diaria.

En el mundo, en muchas ramas de la ciencia existe un gran volumen de información que necesita ser consultada, organizada, y clasificada por especialidades. Esta información especializada conlleva a un lenguaje técnico que no siempre es conocido por los usuarios comunes, además de las regulaciones o estándares que pueden cambiar en diferentes áreas del mundo. El estado en que se encuentran las tecnologías de la comunicación permite que más personas en el planeta estén en contacto y obtengan beneficios con el intercambio de la información referente a sus campos de investigación, lo cual a veces genera confusiones o dificultades para entender un término específico que varía según el lugar en que se use.

En la comunidad médica se pone de manifiesto con más fuerza la diferencia de los estándares regulatorios de su vocabulario por la cantidad de clasificaciones, nombres, descripciones y relaciones que pueden tener las enfermedades, síndromes y la sintomatología de los padecimientos. Se necesita gran cantidad de tiempo para la consulta en bibliotecas y una búsqueda que puede ser relativamente larga en los textos localizados en libros que puedan brindar información. Hasta en un medio informatizado puede llevar a la pérdida de tiempo una búsqueda por la gran cantidad de información irrelevante que puede existir con descripciones similares como es el caso de Internet cuando se usan buscadores clásicos como el Google.

El presente tratamiento del lenguaje se relaciona con el enfoque conocido como Reglas de Negocio en un proyecto que intenta modelar dicho enfoque en los servicios de nefrología con el objetivo de mejorar su funcionamiento. Las Reglas de Negocio son definiciones explícitas que regulan cómo opera un determinado negocio y cómo el mismo es estructurado; se entiende por negocio cualquier tipo de servicio que ofrezca determinada institución. Estas Reglas de Negocio son imprescindibles para el funcionamiento de la empresa o institución, así como para el Sistema de Información que soporta sus procesos.



El problema de este enfoque es que dichas reglas necesitan ser válidas semánticamente para que el sistema de información que soporta sus procesos funcione correctamente. En este punto es necesario dominar la totalidad de los términos especializados que puedan intervenir en las Reglas de Negocio para asegurar su validez semántica.

Para resolver este problema se decide crear un sistema que valide semánticamente las reglas de una manera interactiva con el usuario y que asegure que este no sea necesariamente un diseñador de ontologías o de Reglas de Negocio.

### **Hipótesis de investigación:**

El proceso de validación semántica, como parte de la edición de reglas de negocio para lenguaje médico, puede ser totalmente automatizado haciendo un adecuado uso de ontologías y las nuevas tecnologías de la información.

### **Objetivos Generales:**

- Implementar un sistema para la validación semántica de las Reglas de Negocio basándose en el *Unified Medical Language System* (UMLS).

### **Objetivos Específicos:**

- Estudio de una ontología sobre lenguaje médico especializado para usar en la validación de las Reglas de Negocio
- Identificar las tecnologías y herramientas a utilizar en la construcción del sistema computacional.
- Diseñar e implementar los módulos necesarios para la validación semántica de las Reglas de Negocio.

### **Estructura de la tesis.**

El contenido de la tesis fue dividido en tres capítulos. En el primer capítulo se describe el marco teórico para el trabajo y se caracterizan las herramientas utilizadas para el trabajo con ontologías que nos permiten lograr el objetivo propuesto en el proyecto. En el segundo capítulo se exponen los elementos de diseño e implementación del sistema de validación de Reglas. Finalmente el capítulo tres realiza una descripción detallada que sirve de guía para el uso del sistema de validación semántica.

## ***Capítulo 1***

## **Conceptos y herramientas usadas para la realización del Sistema para la Validación Semántica de Reglas de Negocio.**

En este capítulo se hará una revisión de los conceptos y las características fundamentales de las herramientas usadas en el sistema que se pretende desarrollar. En primer lugar se caracterizan las ontologías y el enfoque de Reglas de Negocio brindando una serie de las principales definiciones y usos de las mismas. Otra temática tratada en el capítulo es la caracterización del sistema que permitirá el desarrollo del Sistema para la Validación Semántica de Reglas de Negocio.

### **1.1 Uso de las Ontologías en las ramas de investigación:**

Las ontologías han ido cobrando cada día más importancia en el ámbito de la computación al irse generalizando su aplicación a diferentes áreas como el procesamiento de lenguaje natural, la interoperabilidad entre sistemas heterogéneos, la indexación de sitios web, el modelado de empresas, la enseñanza, medicina, etc.

En la literatura se encuentran diferentes definiciones sobre ontologías que han ido cambiando en la medida en que se va ganando en conocimiento y se han ido formalizando hasta el punto de aparecer la Ingeniería de Ontologías como una rama más de la ciencia. Se han establecido diferentes clasificaciones atendiendo a varios criterios y se han desarrollado herramientas de diversa índole para la construcción, almacenamiento, consulta y evaluación de las ontologías. Terminando con la aparición de lenguajes para la representación de las mismas como son OWL, RDF, citando algunos de los más significativos.

Las ontologías han llamado la atención en los últimos años en el mundo de la informatización. La investigación sobre las ontologías comenzó a inicio de los años 90 y se ha acelerado extendiéndose a la comunidad de tecnología web en los últimos años.

Muchas personas hablan hoy en día sobre las ontologías. Sin embargo, pueden existir algunos conceptos erróneos sobre qué es una ontología, qué es la ingeniería ontológica, cuan útil puede ser la utilización de las mismas. En los siguientes epígrafes de la tesis se abordan temas acerca de estas ideas.

#### **1.1.1 Concepto de ontología.**

Las ontologías hacen posible una semántica para construir los metadatos que sirven para la estructuración del contenido. Un metadato no es más que un dato estructurado sobre

la información, o sea, información sobre información, o de forma más simple, datos sobre datos. Los metadatos en el contexto de la Web, son datos que se pueden guardar, intercambiar y procesar por medio del ordenador. Están estructurados de tal forma que permiten ayudar a la identificación, descripción, clasificación y localización del contenido de un documento o recurso Web y por tanto, también sirven para su recuperación. Una ontología es una especificación de una conceptualización, esto es, un marco común o una estructura conceptual sistematizada y de consenso no sólo para almacenar la información, sino también para poder buscarla y recuperarla. Una ontología define los términos y las relaciones básicas para la comprensión de un área del conocimiento, así como las reglas para poder combinar los términos para definir las extensiones de este tipo de vocabulario controlado. Se trata de convertir la información en conocimiento mediante unas estructuras formalizadas (las ontologías) que referencien los datos, por medio de los metadatos, bajo un esquema común normalizado sobre algún dominio del conocimiento. Los metadatos no sólo especificarán el esquema de datos que debe aparecer en cada instancia, sino que también podrán contener información adicional de cómo hacer deducciones sobre ellos, es decir, cómo establecer axiomas que podrán, a su vez, aplicarse en los diferentes dominios del conocimiento almacenado.(Mizoguchi, 1997, Gruber, 1993)

### **1.1.2 Tipos de Ontologías**

En 1998, fueron distinguidos tres tipos fundamentales de ontologías(Guarino, 1998):

- Ontologías de un dominio, en las que se representa el conocimiento especializado pertinente de un dominio o subdominio, como la medicina, las aplicaciones militares, la cardiología, etc.
- Ontologías genéricas, en las que se representan conceptos generales y fundacionales del conocimiento como las estructuras parte / todo, la cuantificación, los procesos o los tipos de objetos.
- Ontologías representacionales, en las que se especifican las conceptualizaciones que subyacen a los formalismos de representación del conocimiento, por lo que también se denominan meta-ontologías (meta-level o top-level ontologies).

La ontología que se va a tratar en el presente trabajo es de tipo dominio, se especializa en un campo de la medicina clínica.

### **1.1.3 Otras definiciones.**

Las ontologías se componen de varios elementos cuyas definiciones(Gruber, 1993) se expresan a continuación:

**Conceptos:** son las ideas básicas que se desean formalizar. Los conceptos pueden ser clases de objetos, métodos, planes, estrategias, procesos de razonamiento, etc.

**Relaciones:** representan la interacción y enlace entre los conceptos de un dominio. Suelen formar la taxonomía del dominio. Por ejemplo: subclase-de y parte-de.

**Funciones:** son un tipo concreto de relación donde se identifica un elemento mediante el cálculo de una función que considera varios elementos de la ontología. Por ejemplo, pueden aparecer funciones como: asignar-fecha, categorizar-clase, etc.

**Instancias:** se utilizan para representar objetos determinados de un concepto.

**Reglas de restricción o axiomas:** son teoremas que se declaran sobre relaciones que deben cumplir los elementos de la ontología. Por ejemplo: "Si A y B son de la clase C, entonces A no es subclase de B", "Para todo A que cumpla la condición B1, A es C", etc. Los axiomas, junto con la herencia de conceptos, permiten inferir conocimiento que no esté indicado explícitamente en la taxonomía de conceptos.

## **1.2 Enfoque de Reglas de Negocio.**

Los cambios constantes y rápidos en el ambiente de los negocios son muy comunes en la actualidad. Estos cambios afectan tanto a los negocios como a las aplicaciones de apoyo, por lo que se requieren constantes actualizaciones y adaptaciones. Esto, a su vez, conlleva a la necesidad de que los Sistemas de Información (SI) se adapten y extiendan para que reflejen los cambios producidos, todo se debe realizar, de ser posible, sin la intervención de ningún desarrollador de Sistemas de Información (SI). De acuerdo a varios autores las Reglas de Negocio (RN) tienen un efecto significativo en la escalabilidad de las aplicaciones del negocio.

Las Reglas de Negocio son definiciones explícitas que regulan cómo opera un determinado negocio y cómo el mismo es estructurado; se entiende por negocio cualquier tipo de servicio que ofrezca determinada institución. Estas Reglas de Negocio son imprescindibles para el funcionamiento de la empresa o institución, así como para el Sistema de Información que soporta sus procesos(Group., 2000)

Concretando, la definición formal de una Regla de Negocio se establece como sigue:

**Regla de Negocio:** una sentencia que define o restringe algún aspecto. Con esto se pretende establecer la estructura del negocio o controlar o influir en su comportamiento. Son atómicas en el sentido de que no pueden ser fraccionadas o descompuestas en Reglas de Negocio mas detalladas. Si se redujeran perderían información importante sobre el negocio.

Existen diferentes tipos de Reglas de Negocio, entre las formas más simples se encuentran las tradicionales validaciones que restringen valores permitidos para los datos que se captan; no obstante, no es este el tipo de regla que es de interés en el enfoque de Reglas de Negocio, sino aquellas que constituyen normas, regulaciones, lineamientos, etc., que son representativas del negocio y sensibles a ser modificadas.

En el enfoque de Reglas de Negocio, los usuarios tienen la posibilidad de definir las reglas previo al desarrollo de las aplicaciones, logrando de esta forma una independencia de las aplicaciones que las usan y facilitándose así su mantenimiento; o sea, si cambia una regulación, solo hay que reflejarlo en la definición de la regla y no en los múltiples controles que hacen uso de la misma.

Un patrón de regla se conforma haciendo referencia a otros elementos del modelo, fundamentalmente a objetos del negocio y sus atributos.

Los patrones pueden reflejar la forma de las cosas, los hechos que en la industria o los tipos de problemas que un sistema automatizado pretende negociar.

Debido a que las reglas son usualmente sentencias informales, requieren alguna representación más formal, similar a especificación del lenguaje lógico o notación gráfica. Por esta razón se establecen patrones que permiten escribir cada regla en un lenguaje formal, pero a la vez, cercano al lenguaje del negocio.

### **1.2.1. Patrones para Reglas de Negocio**

Los convenios utilizados para los patrones son los siguientes:

- Paréntesis ( ) encierran un grupo de ítems
- Corchetes [ ] encierran elementos opcionales
- Barra Vertical | separa términos alternativos

# **Capítulo 1**

## **Conceptos y herramientas usadas para la realización del Sistema para la Validación Semántica de Reglas de Negocio.**

- Corchetes angulares < > encierra términos especiales como los definidos en la siguiente tabla.

Es importante destacar que estos símbolos no aparecen en la sentencia de regla sino solamente en el patrón de definición.

A continuación se describen Elementos Variables de los Patrones.

Elemento	Significado
<determinante>	Es el determinante para cada sujeto, por ejemplo: Uno, Una, El, La, Cada, Todos.
<sujeto>	Es una entidad reconocible del negocio, tal como objetos del negocio visibles en el Modelo de Hechos, un nombre de Rol o una propiedad de un objeto. La entidad puede ser cualificada por otros elementos descriptores, tales como la existencia en un estado particular, relacionada con una aplicabilidad específica de la regla con cierta precisión.
<característica>	El comportamiento del negocio que debe tomar lugar o interrelación que debe ser establecida.
<hecho>	La interrelación entre términos identificables en el modelo de hechos. La interrelación puede ser cualificada por otro elemento descriptor relativo a identificar la aplicabilidad de la regla precisada.
<Lista-de-hechos>	Una lista de <hechos> ítems.
<m>, <n>	Son parámetros numéricos.
<resultados>	Cualquier valor, no necesariamente numérico, que tiene algún significado en el negocio. El resultado es además, de forma opcional, el valor de un atributo en un objeto de negocio.
<algoritmo>	Es una definición de la técnica que será usada para obtener el valor del resultado, normalmente expresada usando la combinación de ítems de variables identificables en el Modelo de Hechos unidas a constantes disponibles
<clasificación>	Una definición de un término en el Modelo de Hechos. Esto permite definir típicamente cualquier valor de un atributo, también llamado “estado”, o un subconjunto de objetos en una clase existente.
<lista-enumerada>	Es una lista enumerada de valores. Una numeración abierta indica que la lista puede ser modificada de acuerdo a requerimientos futuros; por ejemplo, una lista de estados de un objeto como lo conocemos actualmente. Una enumeración cerrada indica que los cambios en la lista no están previstos, por ejemplo los días de la

	semana o los meses del año. La identificación resulta muy útil en la posterior implementación.
--	--

**Patrón 1:** Restricción Básica.

Es el más común de los patrones de Reglas de Negocio, establece una restricción sobre un sujeto de una regla.

<determinante> <sujeto> [ no ] (debe |tiene) <característica>

[ ( si | a menos que ) <hecho> ].

< determinante > < sujeto > (puede < característica > solo si <hecho> ) |

( no puede < característica > ).

**Patrón 2:** Lista de Restricción.

Este patrón también restringe al sujeto, pero restringiendo que son uno o mas ítems tomados de una lista.

< determinante > < sujeto > [ no ] (debe |tiene) < característica >

(si | a menos que) como mínimo <m> [ no más de <n> ]

de las siguientes es verdadera:

<lista de hechos>.

< determinante > < sujeto > ( puede < característica > solo si ) |

(no puede < característica > si )

al menos <m> [ y no más de <n> ]

de las siguientes es verdadera:

<lista de hechos>.

**Patrón 3:** Clasificación.

Este patrón establece una definición para un término en el modelo de hechos. Tales términos son equivalentes a términos de corta vida o de uso temporal. Si esta clasificación es permanente se debe reflejar mejor en el Modelo de Hechos.



< determinante > < sujeto > [ no ] es definido como <clasificación>

[ ( si | a menos que ) <hecho> ].

< determinante > < sujeto > [ no ] debe ser considerado como < clasificación >

[ ( si | a menos que ) < hecho > ]

**Patrón 4: Cálculo.**

Es patrón establece una interacción entre términos en el Modelo de Hechos suficientes para realizar los cálculos o el establecer un valor. Note que esto es similar al patrón de clasificación, en cuanto a obtención de valores temporales. Se usa tanto “es definido como” y “debe ser calculado como”

< determinante > < resultado> es definido como <algoritmo>.

< determinante > < resultado > = <algoritmo>.

**Patrón 5: Enumeración.**

Este patrón establece el rango de valores que pueden ser legalmente tomados por un término en el Modelo de Hechos.

< determinante > < resultado > debe ser elegido desde la siguiente lista

[ abierta | cerrada ]:

<lista de valores>

**1.2.2 Forma moderna de implementación de Reglas de Negocio.**

Dentro del contexto de las investigaciones de Reglas de Negocio, estas últimas representan un meta-elemento importante que necesita ser capturado y formalizado separadamente de otros elementos, conformando así la arquitectura de la aplicación. Actualmente es posible el desarrollo de importantes herramientas, basadas en tecnologías orientadas a Reglas de Negocio. Teniendo en cuenta cómo ellas representan, implementan y ejecutan las Reglas de Negocio, son clasificadas como sigue:

**Herramientas independientes de la base de datos:** las Reglas de Negocio son independientes de la base de datos, aun cuando son generadas y administradas automáticamente por herramientas desarrolladas y no por herramientas de la base de datos.

**Herramientas basadas en servicios:** son las Reglas de Negocio creadas por herramientas desarrolladas que se encuentran en los servicios de aplicaciones y residen en el servidor de aplicaciones.

**Sistemas basados en reglas:** intentan especificar restricciones sobre determinados elementos de datos o tablas, orientados a obtener la lógica de negocio al más alto nivel y las reglas asociadas a diferentes situaciones.

### **1.3 Sistema de Lenguaje Medico Unificado (UMLS)**

El *Unified Medical Language System (UMLS)* o Sistema de Lenguaje Medico Unificado está conformado por tres Fuentes de Conocimiento, pueden ser usadas de forma separada o juntas(Olivier Bodenreider).

Estas fuentes son:

- Metathesaurus: más de un millón de conceptos biomédicos desde cien fuentes de vocabularios distintas.
- Red Semántica o *Semantyc Network*: 135 categorías y 54 relaciones definidas entre estas categorías.
- *SPECIALIST Lexicon & Tools*: información lexicográfica y herramientas para el procesamiento del lenguaje.

El *UMLS* comienza a desarrollarse en 1986, por el *National Library of Medicine (NLM)*. Es un proyecto complementario para los Sistemas de Administración Integrados para la Información Académica (*IAIMS*, por sus siglas en inglés) y es un esfuerzo para sobrepasar dos barreras para la extracción efectiva de información legible de las máquinas.

Las barreras son(Olivier Bodenreider):

- La variedad de maneras en que los mismos conceptos son expresados en diferentes códigos y por personas diferentes.

- La distribución de la información útil acerca de bases de datos y sistemas discordes.

El *UMLS* es un proyecto para desarrollar y distribuir Bases de Conocimiento electrónicas multiusos junto a los programas léxicos asociados a ellas para los diseñadores de sistemas.

Las Fuentes de Conocimiento son usadas para superar:

- Disparidades del lenguaje y formatos del mismo.  
Ejemplo: *atrial fibrillation, auricular fibrillation, af*;
- Disparidad en la granularidad y perspectiva.  
Ejemplo: *Contusion, hematoma, bruise*.

Entre los usos del UMLS están:

- Extracción de información.
- Construcción de tesauros.
- Procesamiento de lenguaje natural.
- Indexación automática.
- Archivos de salud electrónicos (EHR, siglas en inglés).
- Mecanismos de distribución para estándares regulatorios (Ej.: HIPAA, CHI, PHIN, etc.).

En la práctica, el UMLS, consiste de tres partes:

- Base de datos. Es la serie de datos relacionales guardados en ficheros.
- Interfaces. *Web Interfaces: Knowledge Source Server(UMLSKS). APIs(Java y XMLs-based)*.
- Aplicaciones. *Lvg* (programas lexicográficos), *MetamorphoSys* (Instalación y personalización).

El UMLS esta disponible como ficheros de datos en tres conjuntos separados de ficheros relacionales, herramientas entre las que se cuentan el MetaMorphoSys,

exploradores de ficheros RRF y programas léxicos. Es importante aclarar que este proyecto del UMLS no es una interfaz final de usuario.

### **1.3.1 Caracterización del Sistema de Lenguaje Médico Unificado.**

El Sistema consta de tres partes o servidores(Dachel Kleinsorge):

#### **Metathesaurus:**

- Fuentes de vocabulario.
- Conceptos.
- Relaciones, Atributos.

#### **Semantic network:**

- Tipos semánticos (Categorías).
- Relaciones Semánticas.

#### **Recursos Lexicos:**

- *SPECIALIST Lexicon.*
- Herramientas léxicas.

### **1.3.2 Metathesaurus**

El *Metathesaurus* forma la base del UMLS y está comprendido por un millón de conceptos biomédicos y cinco millones de nombres de conceptos, todos los cuales están por encima de 119 vocabularios controlados en 17 idiomas y sistemas de clasificación usados en los archivos de pacientes, datos de salud bibliográficos, administrativos y bases de datos de texto completas. Algunos ejemplos de los vocabularios controlados son: el ICD-9-CM, SNOMED CT, LOINC, y RxNORM. El propósito del *Metathesaurus* es proporcionar una base de contexto y relaciones de inter-contexto entre estos varios sistemas codificados y vocabularios para proporcionar una base común de intercambio de información entre la variedad de bases de datos clínicas y sistemas.

El *Metathesaurus* está organizado por conceptos o significados. Cada concepto tiene atributos específicos que definen el significado. Se unen los conceptos idénticos o casi idénticos junto con el contexto jerárquico de los vocabularios diferentes y las relaciones entre ellos son explicadas y representadas.

El alcance del *Metathesaurus* está determinado por el alcance de la fuente de los vocabularios. El propio *Metathesaurus* es producido por el procesamiento automatizado de una versión en lenguaje de maquinas de la fuente del vocabulario, seguido por la intervención humana de edición y revisión. Un individuo puede obtener una copia del *Metathesaurus*, pero es proyectado para ser usado principalmente por diseñadores del sistema porque es un recurso multi-propósito para el UMLS entero.

Puede proporcionar la información a un programa del software para crear los nuevos datos, contestar las preguntas de un usuario, permitir al usuario refinar sus preguntas de consulta y ayudarlo convirtiendo el vocabulario natural a los vocabularios uniformes usados por los sistemas de clasificación estandarizados. Puede usarse en las aplicaciones clínicas para consultar las bases de datos y unirse a los archivos de pacientes. Usado convenientemente con la Red Semántica y el *SPECIALIST Lexicon* gana utilidad por los resultados coherentes y el poder correlativo.

Los conceptos en los cuales basa su funcionamiento el *Metathesaurus* se organizan en una estructura básica consistente (ver la figura 1.1). Un primer grupo que contiene el nombre del concepto con su Identificador Único (CUI). Una sección para los nombres sinónimos. Un espacio para los términos con su Identificador Léxico (LUI), seguido de un conjunto de términos normalizados. Una sección de cadenas, con su identificador único (SUI) y con sus distintos nombres de conceptos. Por último, la sección de los átomos que representan la fuente de los conceptos:

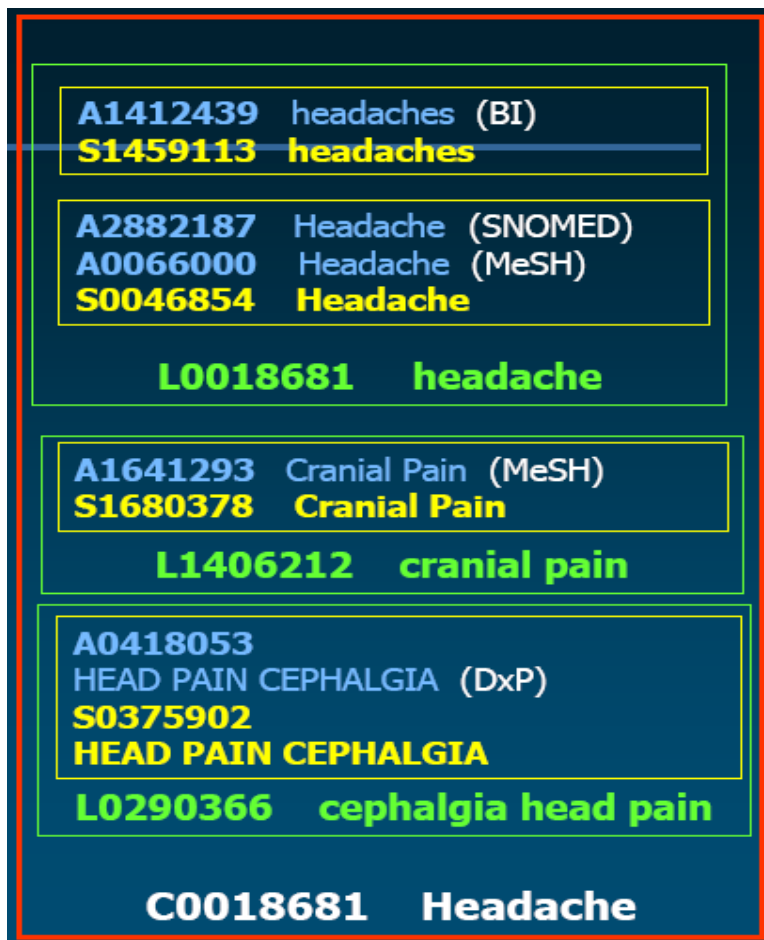


Fig. 1.1. Estructura de concepto.

Metathesaurus agrupa términos por su significado, es decir, agrupa en un mismo concepto varios sinónimos, de los cuales el término preferido por el usuario es escogido mediante la asignación de un identificador único (CUI).

Ej.:

term	source	term type	source ID
Addison's disease	Metathesaurus	PN	
<b>Addison's disease</b>	<b>SNOMED CT</b>	<b>PT</b>	<b>363732003</b>
Addison's Disease	MedlinePlus	PT	T1233
<b>Addison Disease</b>	<b>MeSH</b>	<b>PT</b>	<b>D000224</b>
Bronzed disease	SNOMED Intl	SY	DB-70620
<b>Primary Adrenal Insufficiency</b>	<b>MeSH</b>	<b>EN</b>	<b>D000224</b>
Primary hypoadrenalism syndrome, Addison	MedDRA	LT	10036696

C0001403

Addison's disease

11

Fig. 1.2. Términos agrupados por su significado.

Estos términos quedan agrupados por significados en una estructura de conceptos definida como se muestra en la figura 1.3:

Concept C0001621	Term L0001621	S0011232 Adrenal Gland Diseases S0011231 Adrenal Gland Disease S0000441 Disease of adrenal gland S0481705 Disease of adrenal gland, NOS S0220090 Disease, adrenal gland S0044801 Gland Disease, Adrenal	[...]
	Term L0041793	S0860744 Disorder of adrenal gland, unspecified S0217833 Unspecified disorder of adrenal glands	
	Term L0161347	S0225481 ADRENAL DISORDER S0627685 DISORDER ADRENAL (NOS)	[...]
	Term L0181041	S0632950 Disorder of adrenal gland S0354509 Adrenal Gland Disorders	[...]
	Term L0368399	S0586222 Adrenal disease S0466921 ADRENAL DISEASE, NOS	[...]
	Term L1279026	S1520972 Nebennierenkrankheiten	GER
	Term L0162317	S0226798 SURRENALE, MALADIES	FRE

Fig. 1.3 Grupo de términos parecidos.

Toda esta información se guarda en ficheros relacionales de texto enriquecido en los que cada registro se computa en una sola línea, sus campos de propiedades, atributos y llaves de identificación están separados por el carácter “|”. Son varios ficheros que interactúan entre sí para formar un todo en la ontología.

Los ficheros relacionales mencionados están agrupados en tres grupos principales:

- Conceptos (MRCONSO).

Sinónimos y términos cercanos a la sinonimación.

- Códigos.

Identificadores Únicos de Concepto (CUI). Tipos Semánticos adicionados.

Relaciones y trazados de las mismas (MRREL, MRHIER, MRMAP, MRSMAP).

- Conceptos relacionados con otros conceptos.

Las fuentes de jerarquías están representadas.

Los atributos de las relaciones especifican los tipos y fuentes de las mismas.

Mapas simples y complejos de las relaciones.

Atributos (MRSAT, MRDEF).

#### Ficheros de metadatos

Otros ficheros no menos importantes son los ficheros de metadatos que utiliza el Metathesaurus, historiales e indexado de términos que se muestran a continuación.

- MRSAB: Contiene los fuentes (metadatos).
- MRFILES: Contiene la relación de los ficheros.
- MRCOLS: Contiene los elementos de datos.
- MRDOC: Valores finitos de datos seleccionados.
- MRRANK: Precedencia de fuentes/términos.

#### Ficheros de intercambio

Se usan para interactuar con el sistema, para los casos de ordenar y borrar identificadores CUI, LUI. MRCUI.RRF, para los identificadores únicos de conceptos (CUI) retirados de la ontología y los índices de nombres conceptos.

#### **1.3.3 Red Semántica.**

Las redes semánticas son esquemas de representación de conocimiento involucrando nodos y eslabones (arcos o flechas) entre los nodos. Los nodos representan objetos o conceptos y los eslabones representan las relaciones entre los nodos. Esta representación gráfica ayuda a entender las relaciones de conceptos.

La Red Semántica es una de tres fuentes de conocimiento usadas para facilitar el uso del UMLS. Incluye los tipos semánticos y las relaciones semánticas, estas últimas existen



entre los tipos semánticos. Hay 135 tipos semánticos y 54 relaciones. La red se diseña para categorizar los conceptos en el *Metathesaurus* y proporcionar las relaciones entre los diferentes conceptos que lo conforman. Una vez que un concepto de *Metathesaurus* se establece, se conecta al tipo semántico más específico de la Red Semántica.

Los 135 tipos semánticos están organizados en dos categorías de jerarquía: entidad o evento: Enfermedad de Addison (concepto); Tipo Semántico: Enfermedad o Síndrome. Entre las relaciones mas importantes de los tipos semánticos están las relaciones jerárquicas (“isa”) y asociativas (“causes, causa”).

Ej.: Virus “causa” enfermedad o síndrome.

Hay agrupaciones mayores de tipos semánticos, incluyendo los organismos, las estructuras anatómicas y las funciones biológicas. Los eslabones o relaciones entre los tipos semánticos mantienen la estructura de la red y muestran las relaciones importantes entre las agrupaciones de tipos semánticos y los conceptos. La relación primaria entre los tipos semánticos es la relación "isa" (Fig. 1.4). Esta relación primaria establece una jerarquía para decidir el tipo semántico más específico para asignar a un concepto del *Metathesaurus*.

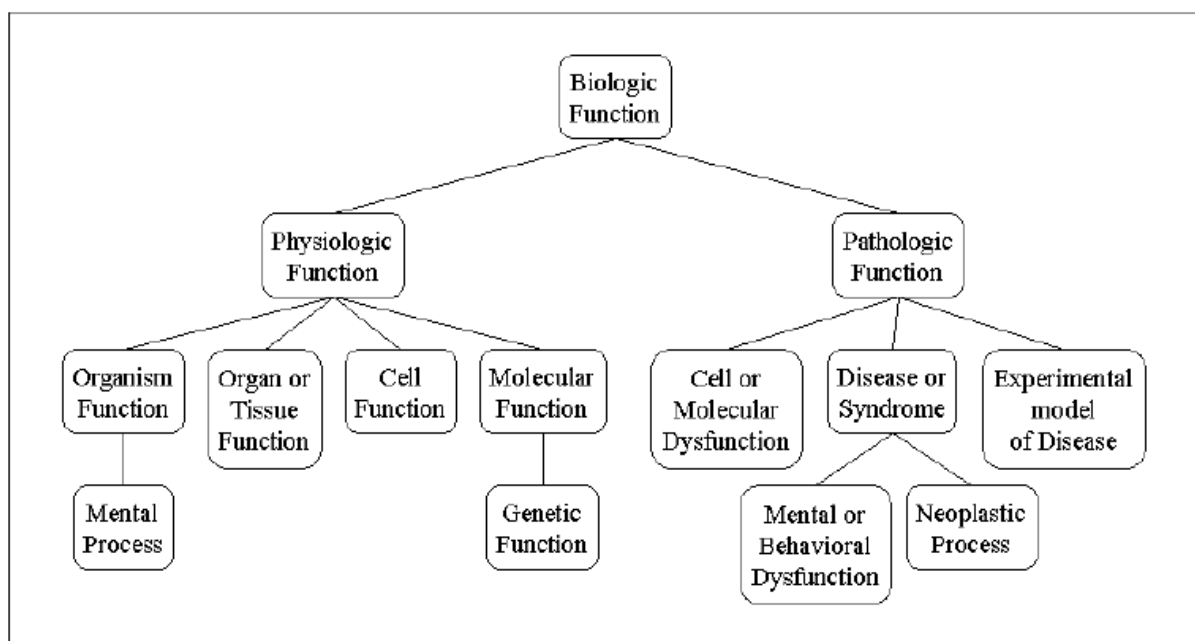


Fig. 1.4 Relación semántica “isa”. (Función Biológica).

La Red Semántica tiene cinco categorías de relaciones no-jerárquicas, categorías llamadas relaciones de asociación. Estas relaciones también pueden nombrarse como relaciones “non-isa” y su red está definida como se muestra en la figura 1.5.

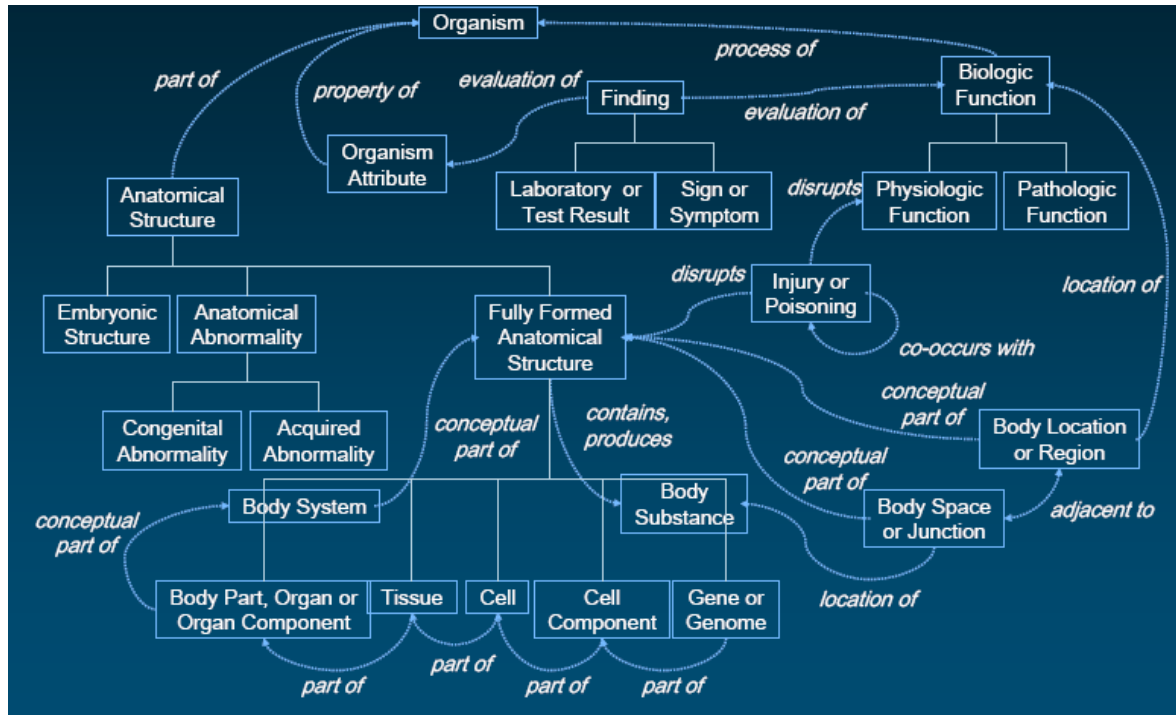


Fig. 1.5 Relación asociativa “non-isa”.

Las relaciones semánticas pueden heredar desde otras relaciones. Un ejemplo de ello es la relación “localizado en” que hereda desde otra relación semántica en el Metathesaurus sin alterar los restantes tipos semánticos (Fig. 1.6).

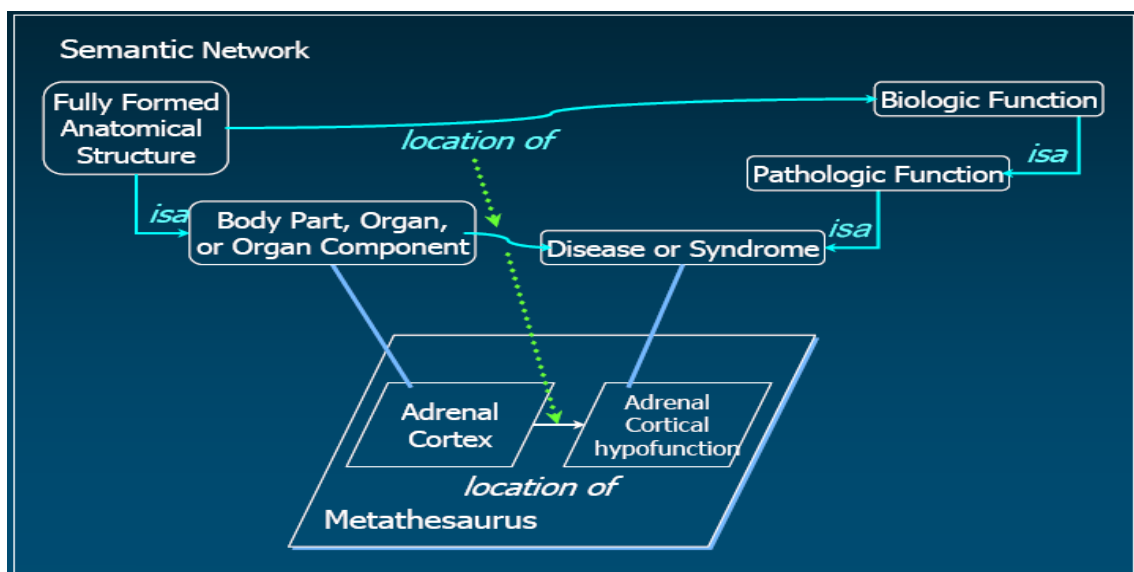


Fig.1.6 Relación “*location of*” hereda semánticas.

La herencia entre las semánticas no es posible para todas las relaciones semánticas, en el caso de la relación “*causa*” se comprueba que no es posible una herencia entre las semánticas. (Fig. 1.7).

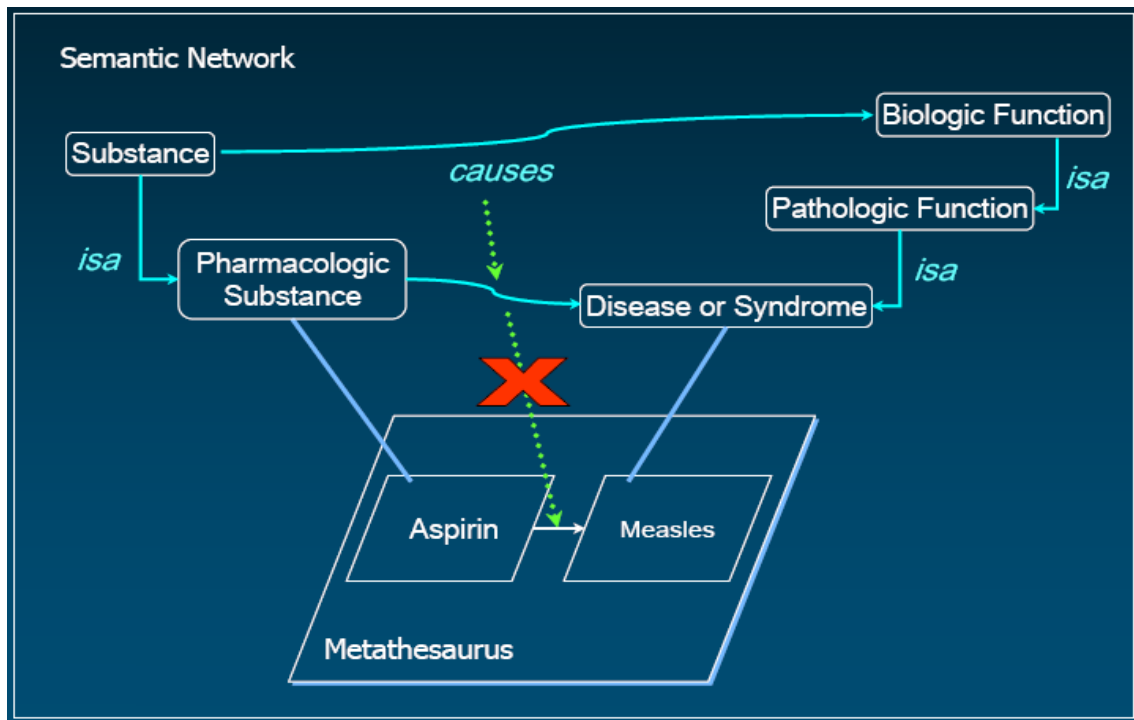


Fig. 1.7 Relación “*causes*” no hereda semánticas.

Las mayores categorías de relaciones semánticas son:

- “físicamente relacionadas a”.
- “espacialmente relacionadas a”.
- “temporalmente relacionadas a”.
- “funcionalmente relacionadas a”.
- “conceptualmente relacionadas a”.

La información del tipo semántico incluye un identificador, su jerarquía, su definición y sus relaciones asociadas.

Las relaciones de asociación dentro de la Red Semántica son muy débiles, ellas capturan algunas relaciones a lo sumo, es decir, capturan el hecho de que algún caso del primer tipo semántico puede conectarse por la relación saliente a algún caso del segundo tipo

semántico. Expresado de otra manera, capturan el hecho de que una afirmación correlativa correspondiente es significativa (aunque no necesita ser verdad).

Tener una Red Semántica sirve como una categorización de alto nivel asignada a cada concepto del Metathesaurus, independientemente de la posición en que esté ubicada la fuente de las jerarquías entre ellos.

Las relaciones semánticas se tratan a partir de ficheros de datos ASCII, agrupados en un grupo que interactúa entre sí.

Algunos de estos ficheros se enumeran a continuación:

- SRDEF: Contiene la información básica acerca de las relaciones semánticas y los tipos semánticos.
- SRSTR: Contiene la estructura de la red semántica.
- SRSTRE1: Contiene el conjunto completo de las relaciones heredadas (UIs).
- SRSTRE2: Contiene el conjunto completo de las relaciones heredadas (nombres).
- SRFIL: Contiene la descripción de cada tabla.
- SRFLD: Contiene la descripción de cada campo y tabla en la que fue encontrada.

#### **1.3.4 SPECIALIST Lexicon.**

El SPECIALIST Lexicon es el tercero de las Fuentes de Conocimiento que apoyan el Sistema del Idioma Médico Unificado. El vocabulario inglés común y las condiciones biomédicas son una fuente para el Sistema Especialista de Procesamiento de Lenguaje, así como la información de MEDLINE, y el UMLS Metathesaurus.

Cada entrada contiene información sintáctica (cómo se reúnen las palabras para crear el significado), morfológica (el formulario y estructura) y ortográfica (deletreando). En el *SPECIALIST Lexicon*, programas JAVA ayudan a los usuarios finales a trabajar a través de las variaciones en los textos biomédicos relacionando las palabras por sus partes de lenguaje que pueden ser útiles en búsquedas Web o búsquedas a través de un registro médico electrónico.

En la información sintáctica se almacena la complementación de sustantivos, verbos y la posición para los adjetivos. Para los sustantivos almacena la información de la frase preposicional. En los verbos interesa su clasificación, entiéndase, intransitivo, ditransitivo o transitivo. Con respecto a la morfología está la información de inflexión de sustantivos, verbos y adjetivos y la derivación de los mismos en otras clasificaciones de palabras (verbos $\Leftrightarrow$ sustantivos; adjetivo $\Leftrightarrow$ sustantivo). Ortografía presupone una revisión ortográfica de términos que estén en sus archivos, en variantes ortográficas.

Las entradas pueden ser términos de una-palabra o de múltiples-palabras. Los archivos contienen cuatro partes: el formulario base (es decir "corra" por "correr"); las partes de lenguaje (del cual el Especialista reconoce once); un único identificador; y cualquier variante de la ortografía disponible. Por ejemplo, una pregunta para el "anestésico" devolvería lo siguiente:

```
{base=anaesthetic
spelling_variant=anesthetic
entry=E0008769
cat=noun
variants=reg}
{base=anaesthetic
spelling_variant=anesthetic
entry=E0008770
cat=adj
variants=inv
position=attrib(3)}
```

El SPECIALIST Lexicon está disponible en dos formatos. El formato "registro de la unidad" que puede verse anteriormente, y se comprende de *slots* y rellenos. Un *slot* es el elemento ("es decir la base = " o "deletreando la variante = ") y los rellenos son los valores atribuibles a ese *slot* para esa entrada. El formato "tabla correlativa" no se normaliza todavía y contiene mucha duplicación de datos.

Este servidor posee además una base de datos auxiliar y ficheros de reglas en los cuales se apoya para realizar su trabajo. Se cuentan las bases de datos auxiliares de Sinónimos (*sm.db*), Derivación (*dm.db*, *dm.rules*), Inflexión (*im.rules*), Compuestos Neoclásicos (*nc.db*). Su conocimiento se basa en tablas de contenido, organizadas de una manera

relacional, o sea, se entrelazan entre sí mismas para conformar una fuente de conocimiento y reglas para su funcionamiento.

Algunas de estas tablas son:

- LRAGR: Contiene los convenios del lenguaje.
- LRCMP: Contiene los complementos del lenguaje.
- LRFIL: Contiene los ficheros.
- LRFLD: Contiene los campos.
- LRMOD: Contiene las modificaciones hechas.
- LRNOM: Contiene la nominalización del lenguaje.
- LRPRN: Pronombres.
- LRPRP: Contiene las propiedades.
- LRSPL: Contiene la ortografía.
- LRTRM: Marcas.
- LRWD: Contiene los índices de palabras.

\*Estas tablas están conformadas como registros de una sola línea, con los campos separados por el carácter especial “[ ]”.

#### **1.4 Conclusiones parciales.**

De todo lo anteriormente explicado, podemos concluir que en el enfoque de Reglas de Negocio es necesaria la validación semántica de las mismas. Es conveniente el sistema *UMLS* para el manejo de los términos que estén relacionados con el dominio que se trata, en este caso, el vocabulario médico y es factible el uso de los códigos fuentes libres del *UMLS* para la implementación del analizador semántico.

## ***Capítulo 2***

## **Diseño e implementación del Sistema para la Validación Semántica de Reglas de Negocio.**

En el presente capítulo se abordan los aspectos relacionados con la arquitectura del sistema, exponiendo los diagramas que permiten modelar, construir y documentar los elementos esenciales que conforman el sistema. Se explicará además, qué herramientas y métodos fueron utilizados para la realización de las interfaces.

### **2.1 Diseño del Sistema.**

#### **2.1.1 Arquitectura del editor de Reglas de Negocio. Localización del módulo para la validación semántica de Reglas de Negocio.**

En estos momentos en la Universidad Central de Las Villas “Marta Abreu” (UCLV) está en desarrollo un sistema de edición de Reglas que cuenta con varias herramientas integradas e interrelacionadas entre las que se cuenta el Analizador Semántico implementado en este proyecto.

A continuación se muestra en la figura 2.1 señalado en líneas rojas el lugar que ocupa el Analizador Semántico en la arquitectura del editor de Reglas.

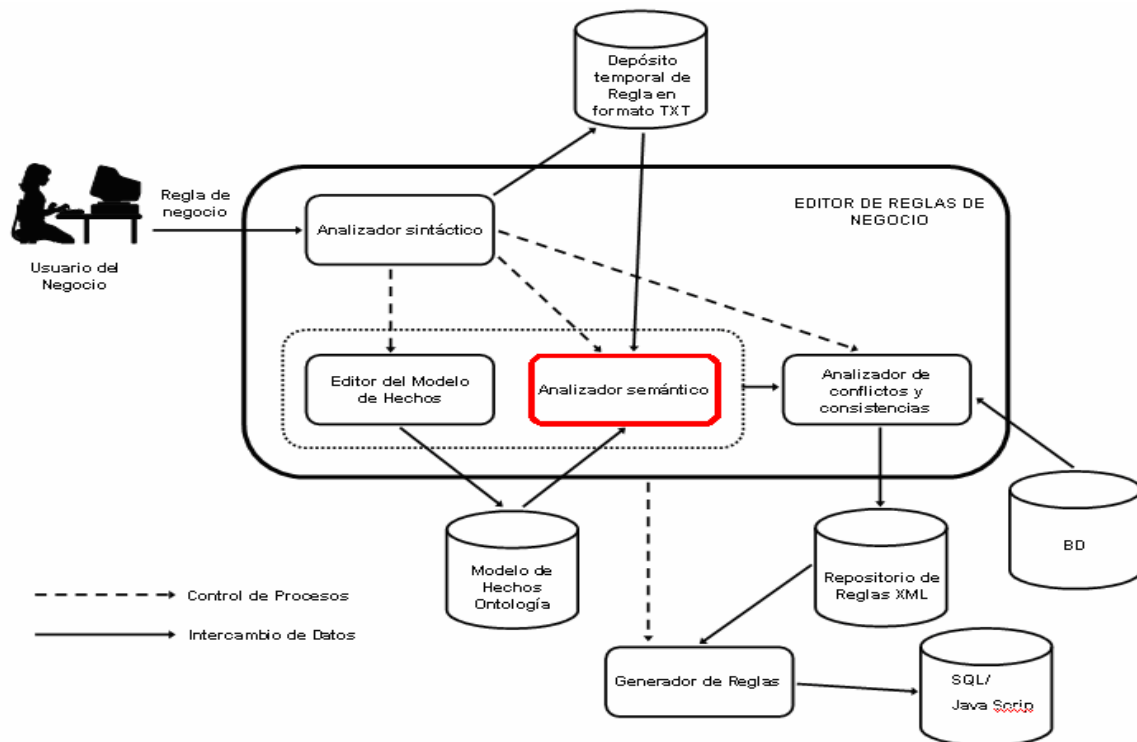


Fig. 2.1 Arquitectura del editor de Reglas de Negocio. Localización del Analizador Semántico de Reglas de Negocio dentro de la Arquitectura del Editor.



### **2.1.2 Análisis y diseño del analizador semántico.**

El análisis y diseño de sistemas es una guía que permite estructurar el proceso de desarrollo de proyectos de software. Se trata básicamente de determinar los objetivos y límites del sistema objeto de análisis, caracterizar su estructura y funcionamiento, marcar las directrices que permitan alcanzar los objetivos propuestos y evaluar sus consecuencias. Se puede afirmar que esta etapa es de vital importancia en el ciclo de vida del proyecto, pues este procedimiento permite reducir al mínimo el riesgo de fracaso de nuevos proyectos. La instalación de un sistema sin la adecuada planeación puede conducir a grandes frustraciones y causar que el sistema sea subutilizado, o peor aún, deje de ser usado al no cumplir con las expectativas que le dieron origen.

A continuación se muestran algunos diagramas que componen el modelado del software.

### **2.1.3 Diagrama de Actividades.**

Un diagrama de actividades representa los flujos de trabajo paso a paso del negocio y operacionales de los componentes en un sistema, lo que es provechoso para entender el comportamiento de alto nivel de la ejecución de un sistema, sin profundizar en los detalles internos. (Torres, 2002).

Diagrama de Actividades en los Anexos. Anexo #1.

### **2.1.4 Actores del Sistema y sus funciones.**

Un actor representa un conjunto coherente de roles que los usuarios de los casos de uso juegan al interactuar con éstos. Normalmente, un actor representa un rol que es jugado por una persona, un dispositivo hardware o incluso otro sistema al interactuar con nuestro sistema. (Alarcón, 2000a).

#### **Actores del sistema y sus funciones:**

##### **Usuario Común:**

- Validar Semánticamente Reglas de Negocio.
- Consultar términos independientes en Ontología (UMLS).

## 2.2 Casos de Uso.

“Un caso de uso es una descripción de un conjunto de secuencias de acciones, incluyendo variantes, que ejecuta el sistema para producir un resultado observable”. (Alarcón, 2000b).

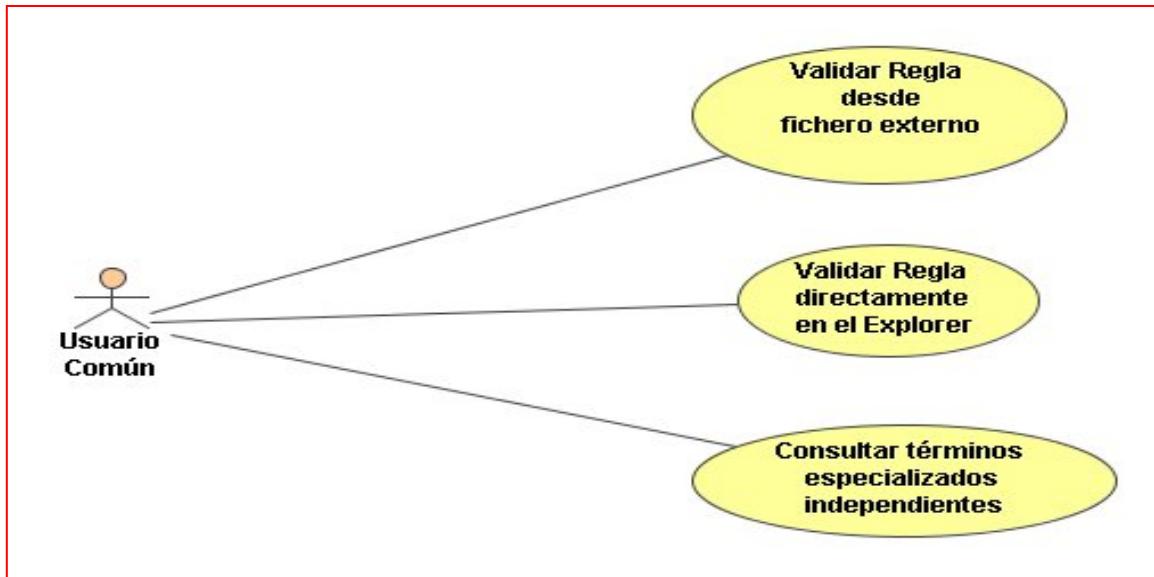


Fig. 2.2. Diagrama de casos de uso del actor **Usuario Común**.

### 2.2.1 Descripción de los Casos de Uso.

Las especificaciones de los casos de uso se hacen de forma tabular mediante una Tabla de Eventos que muestra la interacción entre los actores y el sistema. A continuación describimos algunos casos de uso importantes de cada actor.

#### Descripción del caso de uso “Validar Regla directamente en el Explorer”:

<b>Caso de Uso:</b>	Validar Regla directamente en el Explorer
<b>Actores:</b>	Usuario Común
<b>Propósito:</b>	Validar semánticamente una regla.
<b>Resumen:</b>	El usuario escribe una regla en el área de edición del explorer.

Curso normal de los eventos:	
Acción del actor	Respuesta del sistema
1. Escribir la regla en el área de edición	
2. Dar click en el botón de “Validar”	3. Habilitar el proceso de búsqueda.
	4. Muestra resultados de la búsqueda. Si no es un término asociado al dominio lo notifica.

## **Capítulo 2**

### ***Diseño e implementación del Sistema para la Validación Semántica de Reglas de Negocio.***

5. Decidir si el término de la regla es el adecuado. Si lo es, dar click en el botón “Next” para pasar al próximo término de la regla.	
6. Si no lo es, seleccionar en el área de resultados del explorador el término deseado.	7. Muestra las definiciones y relaciones del término seleccionado.
8. Dar click en el botón “Change” para cambiar un término por otro	9. Cambia el término anterior por el seleccionado por el usuario, edita la regla y la muestra con los cambios para seguir analizándola.
	10. Notifica al usuario si es el último elemento de la regla para que termine el proceso de validación.
11. Dar click en el botón “OK” para guardar la regla en el fichero para su uso posterior.	
12. Repetir el proceso de validación.	
13. Dar click en el botón “Exit”	14. Cierra la aplicación.

#### **Descripción del caso de uso “Validar Regla desde un fichero externo”:**

<b>Caso de Uso:</b>	Validar Regla desde un fichero externo
<b>Actores:</b>	Usuario Común
<b>Propósito:</b>	Validar semánticamente una regla desde la salida de otro sistema.
<b>Resumen:</b>	El usuario selecciona un fichero externo de otro sistema para la validación semántica de las reglas que lo conformen.

<b>Curso normal de los eventos:</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
1. Dar click en el botón “ <i>Validate from File</i> ” para seleccionar el fichero desde donde se hará la validación.	2. Si el directorio existe carga las reglas para validar.
3. Dar click en el botón “ <i>Load Rule</i> ” para cargar las reglas.	4. Muestra la regla en el cuadro de edición.
5. Dar click en el botón de “ <i>Validate</i> ”	6. Habilitar el proceso de búsqueda.
	7. Muestra resultados de la búsqueda. Si no es un término asociado al dominio lo notifica.

## **Capítulo 2**

### ***Diseño e implementación del Sistema para la Validación Semántica de Reglas de Negocio.***

8. Decidir si el término de la regla es el adecuado. Si lo es dar click en el botón “Next” para pasar al próximo término de la regla.	
9. Si no lo es seleccionar en el área de resultados del explorador el término deseado.	10. Muestra las definiciones y relaciones del término seleccionado.
11. Dar click en el botón “Change” para cambiar un término por otro	12. Cambia el término anterior por el seleccionado por el usuario, edita la regla y la muestra con los cambios para seguir analizándola.
	13. Notifica al usuario si es el ultimo elemento de la regla para que termine el proceso de validación.
14. Dar click en el botón “OK” para guardar la regla en el fichero para su uso posterior.	15. Si no es la ultima regla carga la siguiente.
16. Repetir el proceso.	17. Fin del fichero, notifica al usuario.
18. Dar click en el botón “Exit”	19. Cierra la aplicación.

#### **Descripción del caso de uso “Consultar términos especializados independientes”:**

<b>Caso de Uso:</b>	Consultar términos especializados independientes
<b>Actores:</b>	Usuario Común.
<b>Propósito:</b>	Consultar términos independientes del dominio.
<b>Resumen:</b>	El usuario escribe un término en el cuadro de edición del navegador con propósitos de consulta.

<b>Curso normal de los eventos:</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
1. Escribir el término en el área de edición	
2. Dar click en el botón “Search”	3. Se habilita el proceso de búsqueda.
	4. Muestra resultados de la búsqueda. Si no es un término del dominio, se notifica.
5. Selecciona el término a consultar desde el área de resultados.	
6. Repetir el proceso.	
7. Dar click en el botón “Exit”	8. Cierra la aplicación.

### **2.3 Diseño de la interfaz del sistema.**

El diseño de la aplicación posee gran importancia ya que este será el que modele la interacción entre el usuario y el sistema, si bien ya estaba diseñada la interfaz básica, es

necesario que la aplicación se mantenga comprensible, amigable y de fácil aprendizaje para el usuario común, que es el principal actor del sistema de validación. El diseño se concentra en las áreas de búsqueda y visualización del sistema de validación agregando los componentes visuales necesarios para facilitar las acciones a realizar sobre las Reglas de Negocio por el usuario.

### **2.3.1 Herramientas utilizadas para el desarrollo del diseño.**

Para lograr la modificación acorde a las necesidades planteadas al inicio del proyecto se utilizó una interfaz de programación que facilita la programación e inserción de componentes visuales y módulos de funciones y clases de una forma relativamente fácil, siguiendo siempre el método de programación original del sistema.

La interfaz de programación usada para la programación Java de la aplicación es el **NetBeans 5.0**, utilizado para crear el proyecto, programar y compilar los módulos creados para la implementación del sistema de validación.

## **2.4 Implementación del módulo de validación de Reglas de Negocio.**

Para la implementación del módulo de validación de Reglas de Negocio se utilizó como base algunas clases de java ya programadas en el sistema gestor de la ontología (*MetamorphoSys*) mostradas a continuación:

- *WordSearchFinder* (package of Java)
- *WordSearchAction* (class function).
- *RRFBrowserGUI* (package of Java).

A estas últimas se le agregaron funciones y procesos para su adecuación a las necesidades del proyecto para el análisis de las Reglas de Negocio y la validación de todos y cada uno de los términos que aparecen en ellas. Algunas de las funciones son mencionadas a continuación. Serán descritas más adelante.

- *WordSearchAction* (class function).
- *ValidateAction* (class function).
- *ChangeWordAction* (class function).

#### **2.4.1 Herramienta para el manejo de los términos especializados.**

Como fue mencionado anteriormente, para la implementación de este software de validación de Reglas de Negocio se utilizó el *UMLS\MetamorphoSys* aprovechando las ventajas que muestra este sistema para la búsqueda de términos médicos especializados. El sistema brinda una Base de Conocimiento licenciada mundialmente y posee un explorador que cumple con las necesidades básicas de exploración de términos especializados en la Base de Conocimiento. La clase *RRFBrowserGUI* contiene los métodos necesarios para el manejo y exploración de los términos especializados del lenguaje médico y brinda una interfaz cómoda para la visualización de las definiciones de los mismos.

#### **2.4.2 Herramienta para la búsqueda de los términos especializados y la validación de reglas de negocio.**

Para la búsqueda de términos especializados en el antes mencionado explorador (*RRFBrowserGUI*). Fue utilizada la clase *WordSearchFinder* que contiene las funciones para la búsqueda de términos en la Ontología y brinda el ambiente visual necesario para la interacción del usuario con el sistema para la validación de las Reglas de Negocio.

##### **Principales métodos de la clase.**

- *WordSearchAction* (class function).
- *ValidateAction* (class function).
- *ChangeWordAction* (class function).

##### **Clase *WordSearchAction***

La clase *WordSearchAction* hace la función de búsqueda en los ficheros validados de términos especializados del NLM mediante el *thread*, *SearchThread*. La primera acción que toma el *thread* o hilo de ejecución es chequear si en el área de edición existe un texto escrito. Extrae el texto escrito en su cuadro de edición y lo almacena en la variable *search\_field\_words* e inicializa su proceso de búsqueda con los fragmentos o *tokens* de la cadena. Mediante los siguientes fragmentos de código se ilustra la función de chequeo y *tokenización* de la cadena.

```

if (search_field.getText().length() == 0) {
    MMSToolkit.reportError(RRFBrowserGUI.getRRFBrowserGUI(),
        "A search term must be entered.", false);
    RRFBrowserGUI.getRRFBrowserGUI().disableGlassPane();
    return;
}

String[] search_field_words = FieldedStringTokenizer.split(search_field
    .getText(), DELIM);
for (int i = 0; i < search_field_words.length; i++)
    search_field_words[i] = search_field_words[i].toLowerCase();

```

Las funciones **split()** están definidas en las clases del UMLS y fragmentan la cadena de entrada en palabras o respondiendo a comodines y símbolos definidos en el conjunto **DELIM** `(\t-([!@#%&\:;\"',.?/~+=|<>$`^])_!@#%&\:;\"',.?/~+=|<>$`^)` declarado e inicializado en la clase *WordSearchFinder*.

Los resultados de la función **split()** llenan un arreglo (*search\_field\_words*) con los elementos por separado para la búsqueda y tratados con la función de Java **toLowerCase()** para evitar los problemas de comparación que presenta el lenguaje Java en la comparación de caracteres de texto (a “no es igual a” A).

La búsqueda en la ontología se realiza por la función *FindConceptsbyWords()* comparando cada término almacenado en el arreglo *search\_field\_words* con los nombres de conceptos de la ontología y los resultados positivos se muestran en el espacio definido en el software para este propósito. Si no se encuentran coincidencias se manipula un mensaje de error.

### **Clase ValidateAction**

La clase *ValidateAction* modifica a la propia clase *WordSearchAction* para hacer la búsqueda por palabras independientemente de si estas son relacionadas o no en un mismo concepto de la ontología, el objetivo es encontrar los términos que estén relatados en el lenguaje especializado que le da dominio a la ontología y asegurar que la Regla de Negocio que se valide tenga un significado para el área que se trata.

La clase *ValidateAction* recibe un parámetro, *WordParam*, extraído de la Regla de Negocio que se está validando, pasado al hilo de ejecución de la clase para su búsqueda en la ontología.

```
private class ValidateThread extends Thread {  
    /** Constructor */  
    public ValidateThread(String WordParam) { }  
    public void run() {  
        try {  
            String search_field_words = WordParam  
            search_field_words = search_field_words.toLowerCase();
```

Los parámetros que se le pasan a la función *ValidateAction* son extraídos en la ejecución del proceso de validación por el botón *go\_Validate\_Button* que instancia la clase *ValidateAction* aplicando la búsqueda en la ontología.

La variable *WordParam* recibe cada elemento de la Regla de Negocio hasta que llegue a su fin, y lo cede como parámetro a la función de validación. Cuando finaliza el análisis transmite un mensaje de notificación al usuario informándole la culminación del análisis de la regla y poniendo todos los indicadores del analizador semántico a sus valores iniciales para la siguiente regla entrante para su análisis.

### **Clase *ChangeWordAction***

La clase *ChangeWordAction* se usa para la edición de los términos de una regla, significando el cambio de estos por otro término que pertenezca al dominio especificado o preferido por el usuario o especialista encargado de hacer las validaciones de las Reglas de Negocio.

Su principal propósito es sustituir un término por otro, usando para esto los resultados que brindan las funciones de validación y los *tokens* extraídos de cualquiera de las fuentes que brinden las reglas al sistema (el propio software o el editor de reglas)

El siguiente fragmento de código utiliza los valores con que inicia por defecto cada variable del sistema, esta práctica se hace con el propósito de evitar posibles errores en los índices que utiliza la función para realizar los cambios de términos en una regla.



```
Thread t2 = new Thread(new Runnable() {  
    public void run() {  
        Atom[] a = concept_candidates[0].getAtoms();  
        temp = a[0].getStr();  
        search_field_words[0] = temp;  
        temp = "";  
        for (int i = 0; i < search_field_words.length; i++) {  
            temp += search_field_words[i];  
            temp += " ";  
        }  
        search_field.setText(temp);  
        //sacar un parametro para el search field word.  
    }  
});
```

La clase *ChangeWordAction* trata dos parámetros internos del sistema (*Count*, *m*), variables que funcionan como contadores o índices que la sitúan en la palabra que se analiza y que son inicializados con valores neutros, se entiende por valor neutro un número que no influye o esta fuera de los rangos o índices válidos en el sistema. Mediante un chequeo previo de los valores de estos índices se cambian los términos apuntados por ellos, almacenados en las variables, siendo editada la regla posteriormente y mostrada al usuario con los cambios hechos y lista para seguir analizando el resto de ella. El siguiente fragmento de código muestra el uso de las variables que almacenan los índices (*Count*, *m*).

```
Thread t2 = new Thread(new Runnable() {  
    public void run() {  
        Atom[] a = concept_candidates[Count].getAtoms();  
        System.out.println(a[0].getStr());  
        temp = a[0].getStr();  
        search_field_words[m] = temp;  
        temp = "";  
        for (int i = 0; i < search_field_words.length; i++) {  
            temp += search_field_words[i];  
            temp += " ";  
        }  
        search_field.setText(temp);  
        //sacar un parametro para el search field word.  
    }  
});
```

### 2.4.3 Función de validación desde un fichero externo.

La opción de validar una Regla de Negocio desde un fichero externo que puede ser la salida de un editor sintáctico de reglas de Negocio es implementada con el propósito de brindar facilidad al usuario en caso de que el número de Reglas de Negocio a validar sea grande. De una forma automática la función carga la dirección de un fichero texto que contenga las Reglas de Negocio en un formato de texto y organizadas por líneas.

La validación desde un fichero se divide en dos módulos que interactúan y definen una serie de variables y parámetros que usarán las clases anteriormente definidas para la búsqueda y validación de las Reglas de Negocio.

El primer elemento que inicializa parámetros necesarios para la función de validación desde ficheros externos al sistema es el botón: *go\_Validate\_from\_File* que establece la dirección y abre el fichero que contiene las Reglas de Negocio para su lectura por el analizador semántico. En el siguiente fragmento de código se ilustra como está implementada la función.

```
go_validatefromFile_button.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        JFileChooser selector = new JFileChooser();
        int returnVal = selector.showOpenDialog(null);
        if (returnVal == JFileChooser.APPROVE_OPTION) {
            String file = selector.getSelectedFile().getAbsolutePath();
            file = file.replace("\\", "\\\"+\"\\");
            try{
                File file_for_validate = new File(file);
                FileReader fr = new FileReader(file_for_validate);
                BufferedReader br = new BufferedReader(fr);
                String t = br.readLine();
                int i = 0;
                while (t != null) {
                    Array_to_File[i] = t;
                    t = br.readLine();
                    i++;
                }
                flag = true;
            } catch (IOException io){
                io.printStackTrace();
            }
        }
    }
});
```

La función definida en la acción del control (botón *Validate from File*) almacena para el uso del fichero que contiene las Reglas de Negocio la dirección absoluta del mismo en la variable global *file*, posibilitando que el usuario pueda almacenar el fichero en el

lugar que estime conveniente incluso en archivos que sean externos a la computadora donde este instalado el sistema para la validación semántica.

Los ficheros externos deben obedecer a una plantilla predeterminada para el correcto funcionamiento del sistema. La forma en que se almacenan las Reglas de Negocio en el archivo deben tener un formato de texto, sin estilos, y deben estar ordenadas en el archivo de texto en registros de una sola línea, o sea, cada regla debe ocupar una y solo una línea de texto finalizando con el carácter “\n” o “salto de línea” (*return*).

Es válido aclarar que la dirección adquirida es transformada mediante la función *replace* en consecuencia de las diferencias que existen entre los sistemas operativos en cuanto al formato en que se muestran las direcciones del sistema de archivos.

Las Reglas de Negocio son cargadas íntegramente del archivo y almacenadas en un arreglo (*Array\_to\_File*) que facilita su uso y posterior escritura en los ficheros de salida para su uso posterior. Desde la variable (*Array\_to\_File*) el sistema carga mediante la acción del botón *load\_rule\_from\_var* las reglas para que sean visualizadas y analizadas por el usuario. El fragmento de código siguiente muestra la implementación de la acción del control.

```
load_rule_from_var.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent e) {  
        if (ind > Array_to_File.length || Array_to_File[ind + 1] == null) {  
            JOptionPane.showMessageDialog(null, "No more rules in the file",  
                "Notification", JOptionPane.OK_OPTION,  
                JOptionPane.INFORMATION_MESSAGE);  
            ind = -1;  
            flag = false;  
        }  
  
        if (flag) {  
            ind++;  
            search_field.setText(Array_to_File[ind]);  
            flag = false;  
        }  
    }  
});
```

## 2.5 Conclusiones parciales.

De todo lo anteriormente explicado podemos concluir que en la etapa de diseño del analizador semántico se tomaron en cuenta las necesidades y especificaciones del proceso de validación de Reglas de Negocio. Se utilizaron las librerías del *UMLS* para

los procesos de búsqueda y particularmente la clase *WordSearchFinder* que permitió separar los módulos de búsqueda para su modificación en el proceso de implementación del analizador semántico.

## ***Capítulo 3***

## **Guía del Sistema de Validación Semántica de Reglas de Negocio.**

El Sistema de Validación Semántica de Reglas de Negocio está diseñado para asegurar la validez semántica de las reglas que intervienen en el proceso de nefrología para el mejoramiento de su funcionamiento.

**El contenido de este capítulo se estructura de la siguiente forma:**

- El epígrafe 3.1 muestra las características del sistema, los requerimientos mínimos para que funcione correctamente, así como los pasos para la instalación del software que le sirve de base al Sistema de Validación Semántica de Reglas de Negocio.
- El epígrafe 3.2 explica la forma de validación semántica directamente en el explorador de los términos especializados.
- El epígrafe 3.3 explica la forma de validación semántica desde un fichero externo, que es la salida desde otro sistema que conforma el analizador sintáctico.
- El epígrafe 3.4 explica cómo consultar términos independientes del lenguaje médico especializado.

### **3.1 Características generales del Sistema.**

El Sistema de Validación Semántica de Reglas de Negocio es una aplicación que permite el análisis semántico de las Reglas de Negocio que participan en el proceso de nefrología.

Cuenta con un diseño sencillo y de fácil operabilidad acorde a las necesidades y posibilidades de los usuarios que van a interactuar con la aplicación. Aprovechando las potencialidades de la programación en Java y dado que es el lenguaje en que se implementó el software que le sirve de base al Sistema de Validación, se utilizaron las clases implementadas del *MetamorphoSys* y su explorador básico para la implementación del propio Sistema de Validación Semántica, añadiendo al mismo las funciones para el análisis de cada uno de los elementos de las reglas que pasarán al sistema. Para lograr estos propósitos se usó la programación en lenguaje Java mediante el IDE NetBeans 5.0 ganando en rapidez y facilidades para la compilación de los módulos agregados al sistema.

Tiene el sistema tres funciones principales para su uso:

- Validación Semántica de Reglas directamente en el explorador.
- Validación Semántica de Reglas desde un fichero externo.
- Consulta de términos especializados independientes.

Estas funciones interactúan con una ontología la cual tiene como dominio de especialización el lenguaje médico (*UMLS*), que a su vez está administrada por el *MetamorphoSys*.

### **3.1.1 Especificaciones del MetamorphoSys.**

Es necesario tener la instalación del *MetamorphoSys*, este sistema instala y personaliza el *UMLS*. Es un sistema multiplataforma Java de muy fácil uso e instalación.

El software puede ser instalado en máquinas con los sistemas operativos Sun Solaris 8 & 9, Windows XP, NT, 2000, Linux y Macintosh. Para un tiempo aceptable de instalación y para su uso posterior de una manera fluida la máquina debe ser rápida, con un procesador de 1 GHz o más, como mínimo 512 MB de RAM aunque es recomendado que tenga 1 GB, y al menos 20 GB de espacio en disco (*HDD*)(Dachel Kleinsorge).

### **3.1.2 Instalación.**

Para el uso del analizador semántico es necesario instalar el *UMLS* mediante una herramienta distribuida con el propio sistema *UMLS*, se trata del *MetamorphoSys*.

Es una herramienta multiplataforma, es decir, que se ejecuta en casi cualquier sistema operativo, basado en lenguaje de software libre Java.

El asistente de instalación y configuración del *UMLS* (*MetamorphoSys*) instala y personaliza las bases de conocimiento en un dispositivo de almacenamiento local (*HDD*). Personaliza y crea un subconjunto de *Metathesaurus* local en el equipo que se escoja para la instalación y extrae los archivos comprimidos nativos del *metathesaurus*. Es de fácil uso y posee pantallas y etiquetas que guían al usuario a través del proceso de instalación. Esta herramienta se usa para personalizar el subconjunto del *Metathesaurus* y excluir la terminología que puede ser inútil y hasta dañina para ciertas necesidades y propósitos. Cambiando las opciones predefinidas se pueden alterar nombres preferidos dándole más importancia o precedencia para los procesos de búsqueda o alterar la supresibilidad y orígenes específicos de algunos tipos de términos.

El *MetamorphoSys* funciona borrando todos los ficheros relacionales de los vocabularios excluidos por el usuario en la instalación del sistema (átomos, cadenas, relaciones, atributos, trazados), aplica opciones adicionales seleccionadas por el usuario (precedencia o supresibilidad), y produce un conjunto personalizado de ficheros relacionales del *Metathesaurus* reflejando los criterios seleccionados.

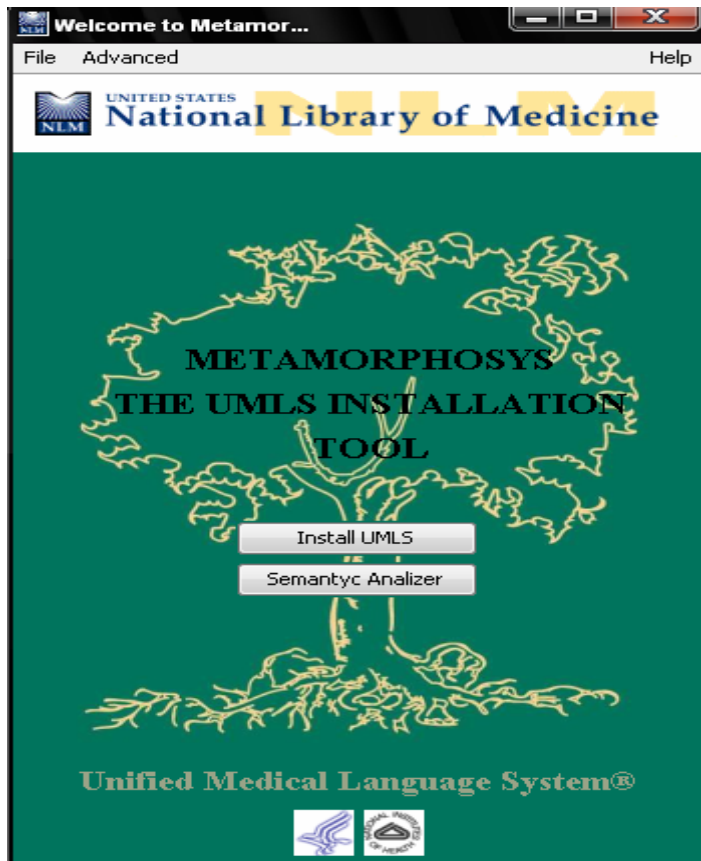


Fig. 3.1 Pantalla de Bienvenida.

Desde la pantalla de bienvenida (Fig. 3.1) el usuario tiene la opción para la instalación del sistema, que ejecutara el asistente para la instalación y configuración de las bases de conocimiento y los elementos que permiten el funcionamiento del *UMLS*. Ver Fig. 3.2.



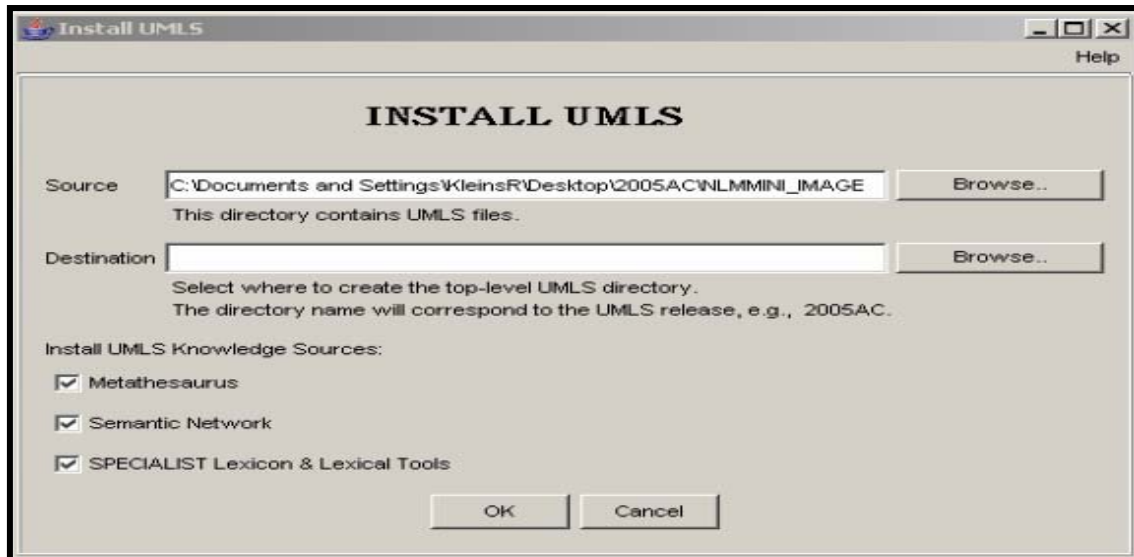


Fig. 3.2 Install *UMLS*.

Seleccionar los elementos a instalar desde esta pantalla (Fig. 3.2), en la cual se puede establecer qué recurso se desea usar. Es recomendable la instalación de los tres elementos del sistema a menos que ya esté en uso alguno de ellos local y previamente configurado. Se debe indicar la localización en que se almacenarán los ficheros que conforman la ontología en los espacios habilitados para ello en la pantalla.



Fig. 3.3 Ficheros de Configuración.

Los nuevos usuarios tienen que seleccionar la opción “*New Configuration*” para la creación de un fichero de configuración personalizado. Si no es un nuevo usuario y ya existe un fichero de configuración del *MetamorphoSys* se puede cargar desde la localización en que se encuentre. Esta pantalla solo aparece si se está instalando el *Metathesaurus*. Para el siguiente paso de la instalación se debe consultar la ayuda del *Wizard* para seleccionar el conjunto por defecto con el que se instalarán las bases de conocimiento del sistema y los requisitos regulatorios que pueden cumplir.



Fig. 3.4 Filtros

Cada uno de estos filtros (Fig. 3.4) habilita una etiqueta de opciones y atributos por su origen, lenguajes, relaciones y tipos semánticos con los que se configurarán las bases de conocimiento del UMLS.

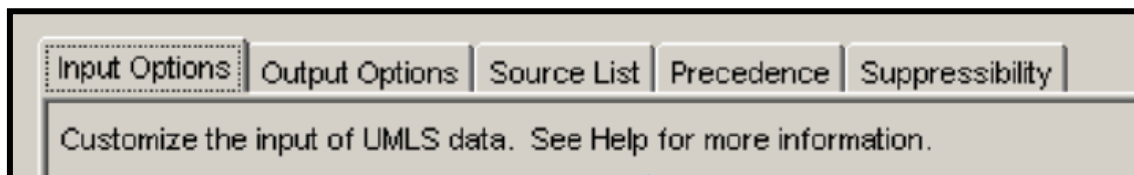


Fig. 3.5 Opciones de entrada.

Las opciones de entrada (Fig. 3.5) son puestas por defecto por el sistema, aunque son editables en caso de que el usuario desee obtener otra versión validada de los estándares regulatorios del vocabulario médico.

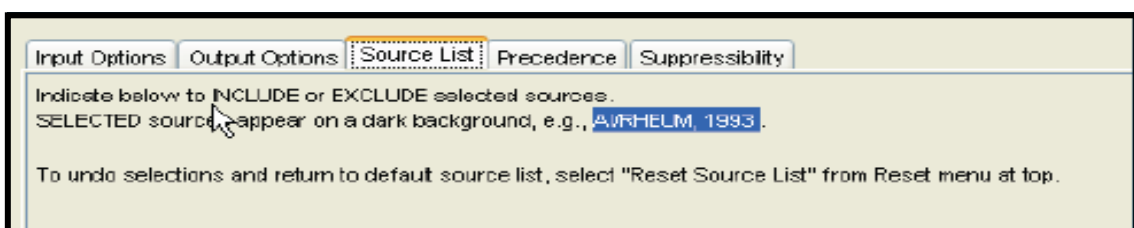


Fig. 3.6 Listas de fuentes del vocabulario.

Full Source Name	Source Abbreviation	Source Family	Language	Level
AIRHEUM, 1993	AIR93	AIR	ENG	0
Alternative Billing Concepts, 2004	ALT2004	ALT	ENG	3
Alcohol and Other Drug Thesaurus, 2000	AOD2000	AOD	ENG	0
Beth Israel Vocabulary, 1.0	BI98	BI	ENG	2
Clinical Clinical Problem Statement System, 1999	CCPSS99	CCPSS	ENG	3
Clinical Classifications Software, 2003	CCS2003	CCS	ENG	0
Current Dental Terminology 2005 (CDT-5), 5	CDT5	CDT	ENG	3
COSTAR, 1989-1995	COSTAR_89-95	COSTAR	ENG	0
Medical Entities Dictionary, 2003	CPM2003	CPM	ENG	2
Physicians' Current Procedural Terminology, Spanish Translation, 2001	CPT01SP	CPT	SPA	3
Physicians' Current Procedural Terminology, 2005	CPT2005	CPT	ENG	3
CRSP Thesaurus, 2004	CSP2004	CSP	ENG	0
COSTART, 1995	CST95	CST	ENG	0
Diseases Database, 2000	DDB00	DDB	ENG	3
German translation of ICD10, 1995	DMDICD10_1995	ICD10	GER	1
German translation of UMDNS, 1996	DMDUMD_1996	UMD	GER	1
DSM-III-R, 1987	DSM3R_1987	DSM3R	ENG	3
DSM-IV, 1994	DSM4_1994	DSM4	ENG	3
DXplain, 1994	DXP94	DXP	ENG	0
Gene Ontology, 2004_12_20	GO2004_12_20	GO	ENG	0
HCPCS Version of Current Dental Terminology 2005 (CDT-5), 5	HCDT5	CDT	ENG	3

Fig. 3.7 Selección de las exclusiones de los orígenes del lenguaje.

Las filas resaltadas en la pantalla (Fig. 3.7) son excluidas del subconjunto del vocabulario especializado usado. Especificando los vocabularios a usar o a excluir se facilita la búsqueda de términos en consecuencia de que la ontología se simplifica y se ordena a partir de los términos preferidos por los usuarios en la zona.

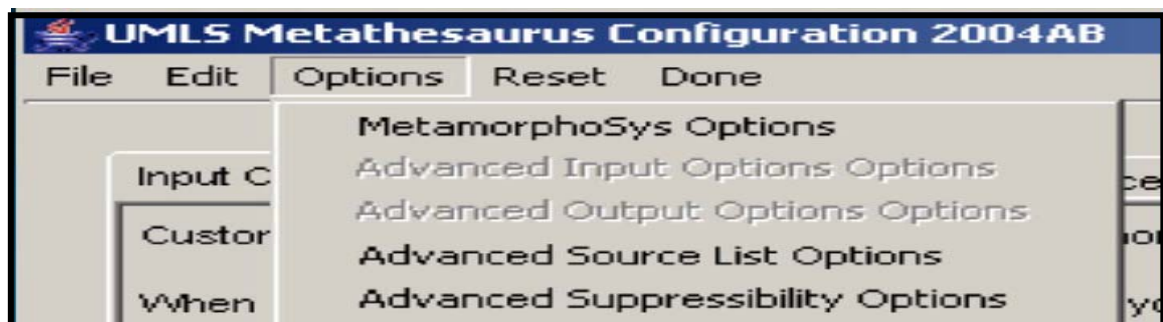


Fig. 3.8 Opciones del MetamorphoSys.

El comportamiento de la lista de orígenes del vocabulario puede ser cambiado por esta pantalla de opciones del MetamorphoSys (Fig. 3.8). Incluso para seleccionar automáticamente los términos relacionados

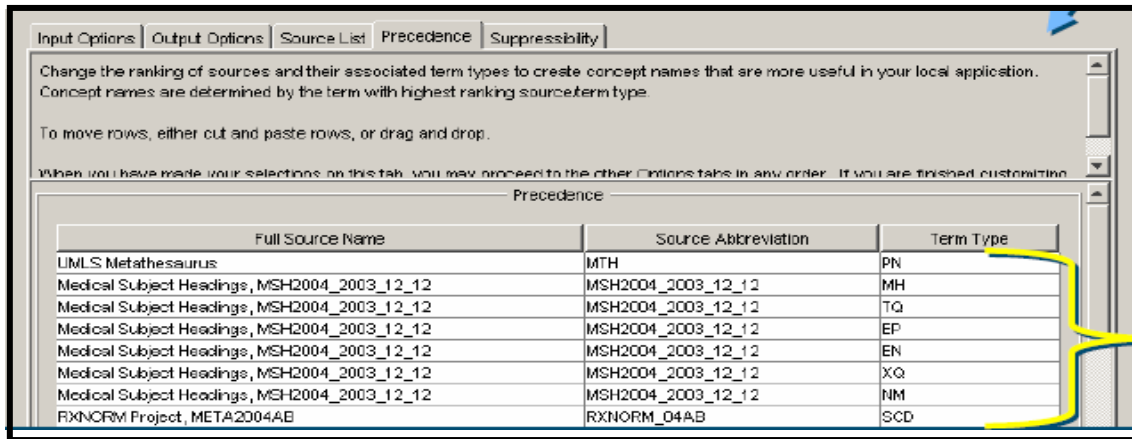


Fig. 3.9 Precedentes.

En la etiqueta *precedence* (Fig. 3.9) se clasifican los nombres por tipos de términos con sus orígenes de vocabulario. Los nombres de la clasificación jerárquica más altos determinan el nombre preferido. Se cortan y pegan las filas para alterar estos nombres preferidos en el vocabulario. Conjugado con la etiqueta *supresibility* se configura la frecuencia de uso de términos que resultan sinónimos o términos que no son usados frecuentemente por los usuarios específicos del sistema.



Fig. 3.10 Done. Iniciando la instalación del UMLS personalizado.

En el menú Done (Fig. 3.10) se completa las opciones de configuración e inicia la instalación del subconjunto del UMLS.

Todos los cambios realizados o problemas en la instalación se reportan en el fichero mmsys.log del MetamorphoSys.

### 3.2 Guía del Analizador Semántico.

Para instalar los módulos del Analizador Semántico es necesario después de la instalación del UMLS copiar la librería mms.jar brindada con los ficheros de instalación del MetamorphoSys en la carpeta \lib. de la instalación del UMLS.

Para acceder al navegador que brinda las opciones de validación semántica se debe ejecutar el UMLS y acceder a la pantalla (Welcome Screen) vista en la figura 3.1 del capítulo en curso. A partir de ella, en la sección inferior, en el botón “Semantyc Analyzer” se lanza el analizador (Fig. 3.11).

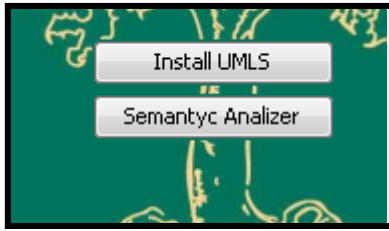


Fig. 3.11 Ejecutar el analizador semántico.

Después de la ejecución del analizador aparece el ambiente en que el usuario realizara los procesos de validación y/o consulta que se muestra en la figura 3.12.

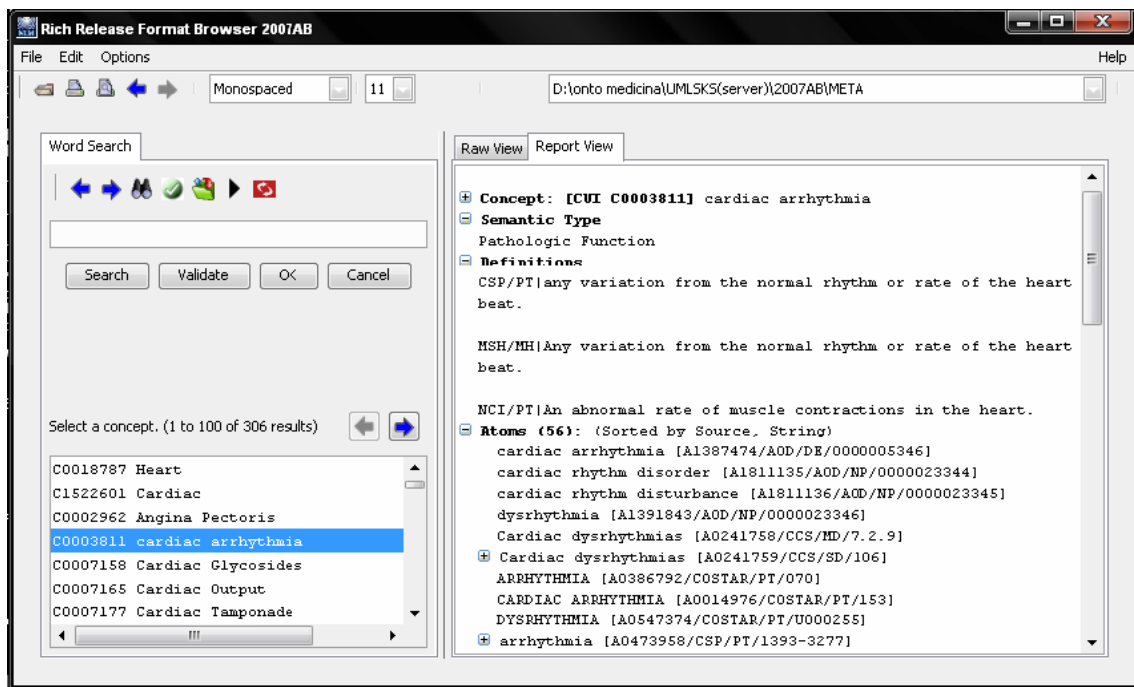


Fig. 3.12 Vista principal del Analizador Semántico.

En la vista principal del analizador semántico (Fig. 3.12) se encuentran los controles básicos que permiten trabajar con la ontología. En la parte superior, cuenta con un menú que controla los aspectos y opciones de configuración del navegador. En el menú File se controla el conjunto de la ontología sobre la que trabaja el navegador. En caso de ser la primera vez que se lanza el navegador es obligatorio que se especifique un directorio RRF válido para el uso del sistema. En las ocasiones subsiguientes en que se use el navegador basta con aceptar el diálogo de notificación al usuario que presenta el navegador.

El navegador mostrará los resultados de las búsquedas de términos, ya sea para la validación de Reglas de Negocio o términos especializados independientes, en las zonas

derecha e inferior izquierda de la aplicación con los nombres y definiciones de los conceptos hallados en la ontología.

### **3.2.1 Validación en el explorador.**

Este proceso permite la validación semántica de Reglas de Negocio directamente escritas por un usuario en lenguaje natural en el área de edición del analizador semántico en caso de tener la facilidad de que sean relativamente pocas reglas a validar. El actor que interactúa con este proceso es el **usuario común** después de haber ejecutado el ambiente del analizador semántico. Las funciones que él puede realizar dentro de esta interfaz se muestran a continuación.

Para poder acceder a las ventajas de la validación semántica de sistema lo primero que debe hacer el usuario es ejecutar la interfaz desde la ventana Welcome Screen (fig.3.14), si se ha ejecutado antes el navegador se le notificará al usuario si desea mantener el último resultado que se consultó en el navegador (Fig. 3.13). Se recomienda que después de haber utilizado por primera vez el navegador del *UMLS* se escoja la opción “Sí”, para la facilitación de la consulta de la ontología especializada. A continuación se muestra el mensaje que se brinda al usuario al comenzar a trabajar con el navegador. Este diálogo solo es mostrado si ya se ha usado más de una vez el navegador para la validación semántica.

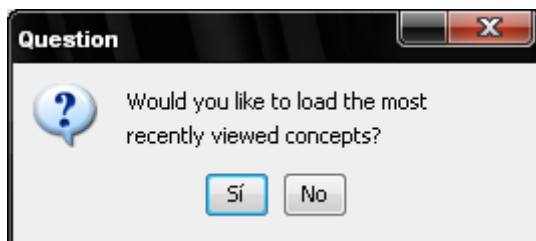


Fig. 3.13 Diálogo de confirmación para el conjunto de ficheros de la ontología.

En el caso de la validación desde un fichero externo se muestra el mismo mensaje de confirmación para establecer el conjunto de ficheros RRF válidos para el navegador.

En la sección superior izquierda del navegador para la validación semántica se encuentra una barra de herramientas con todas las opciones de validación y búsqueda habilitadas solamente cuando se especifica el conjunto RRF válido para su funcionamiento. Debajo de esta barra de herramientas se encuentra el cuadro de edición que sirve de área de intercambio con el sistema en tanto se estén validando e intercambiando términos de una regla del negocio. Ver Fig. 3.12.

### Efectuar la validación de una regla.

Para la validación de una regla del negocio, el usuario debe asegurarse de tener el conjunto RRF abierto.

En el cuadro de edición el usuario escribe la regla del negocio que desee validar en un lenguaje natural atendiendo a la conformación sintáctica de la misma. El sistema no se responsabiliza con la sintaxis de la regla, está implementado solo para la validación semántica.

Si el usuario no escribe algún texto en el cuadro de edición el sistema mostrará un mensaje de error notificando al usuario que debe escribir algo en la entrada del navegador. (Fig.3.14)

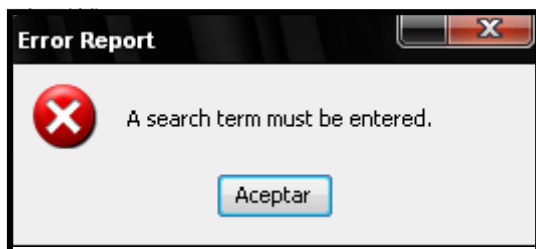


Fig. 3.14 Notificación de error. El usuario no ha escrito ninguna regla o término.

Luego de escribir la regla en el cuadro de edición (Fig. 3.15) se pueden realizar varias acciones sobre la regla entrada.

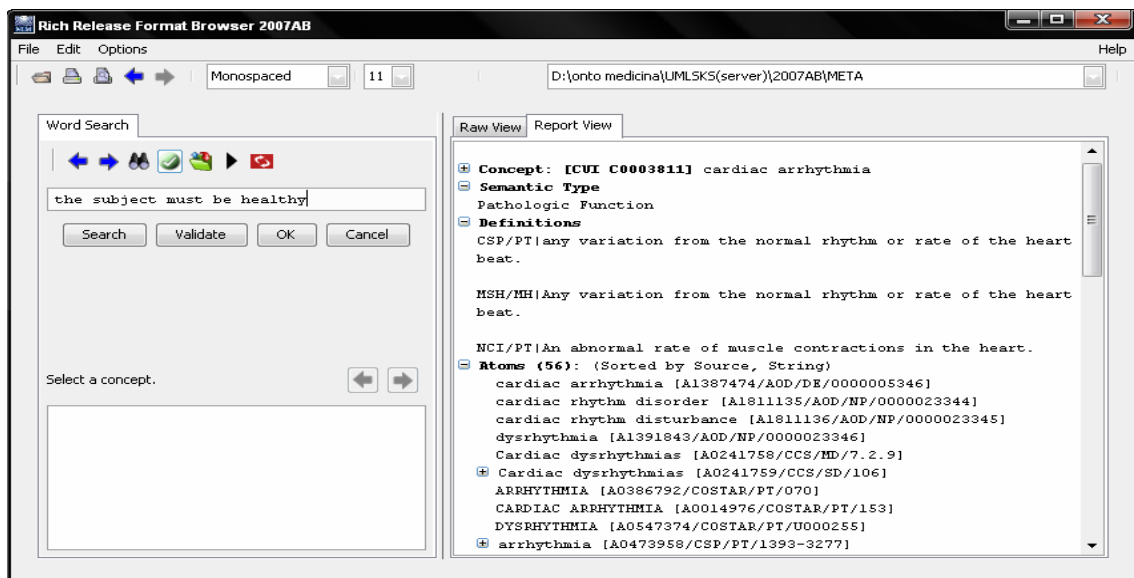


Fig. 3.15 Regla de Negocio escrita en el navegador.

Como ejemplo se insertó la regla: *The patient must be healthy.*

En la barra de herramientas, en la sección de validación y búsqueda de términos independientes, seleccionando el botón correspondiente (Fig.3.16), se inicia la validación de la regla que está en el editor.

De izquierda a derecha se encuentran los botones para avanzar o retroceder en la regla que se esté analizando, estos son: el botón para la búsqueda de términos independientes, el botón para la validación de reglas, el botón para la validación desde un fichero externo y por ultimo el botón para el cambio de términos en la regla con el proceso de edición.



Fig. 3.16 Validación semántica de reglas.

Después de seleccionada la acción de validación se comienza el análisis de cada término que conforma la regla para su validación semántica.

Para el cambio de los términos que no se encuentren en el dominio especializado del lenguaje médico o no estén considerados entre los estándares manejados en la región, se selecciona en la sección para la visualización de los resultados de las búsquedas el término preferido, mediante un doble clic. A través del botón “Change” (Fig. 3.17) localizado en la barra de herramientas se intercambia en la regla el término que esté actualmente en análisis por el seleccionado en los resultados de la búsqueda (Fig. 3.18).



Fig. 3.17 Botón para el cambio de términos en una regla.



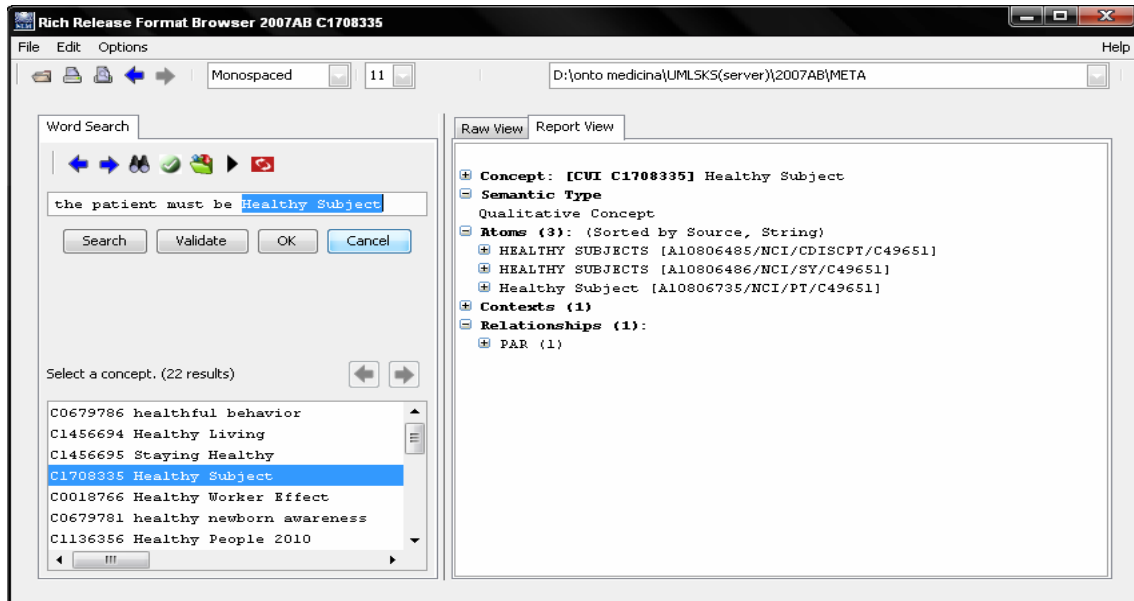


Fig. 3.18 Cambio de términos en una Regla de Negocio.

Al terminar con el análisis de la regla, mediante el botón “OK” se almacena la regla en un fichero para su uso posterior (Fig.3.19).

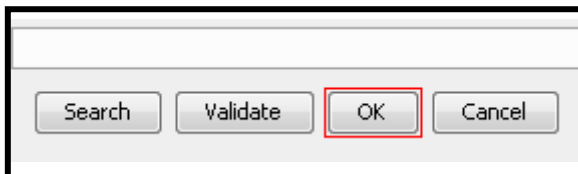


Fig. 3.19 Fin del proceso de validación.

El resultado del proceso para la regla de ejemplo descrita anteriormente sería:

Antes: *The patient must be healthy.*

Después: *The patient must be a healthy subject.*

### 3.2.2 Validación desde fichero externo.

Este proceso permite la validación semántica de Reglas de Negocio almacenadas en un fichero externo que se puede conformar a mano, o puede constituir la salida de otro sistema editor de reglas de negocio que asegura su validez sintáctica o léxica. En caso de conformar el fichero a mano, se debe asegurar, al igual que si se escribiera directamente en el navegador, su validez semántica.

Las funciones para la validación de reglas desde un fichero externo están sujetas a las mismas sugerencias que la validación directamente en el navegador. Al ejecutarse el navegador se debe especificar el conjunto RRF a usarse, ya sea abriéndolo desde el menú File, o confirmando el uso de los últimos términos buscados.

En la barra de herramientas se selecciona el botón correspondiente para la validación desde un fichero externo (Fig. 3.20).



Fig. 3.20 Validar desde un fichero externo.

Este control localizado en la barra de herramientas obtiene la dirección del fichero texto que contiene las Reglas de Negocios para validar semánticamente (Fig. 3.21).

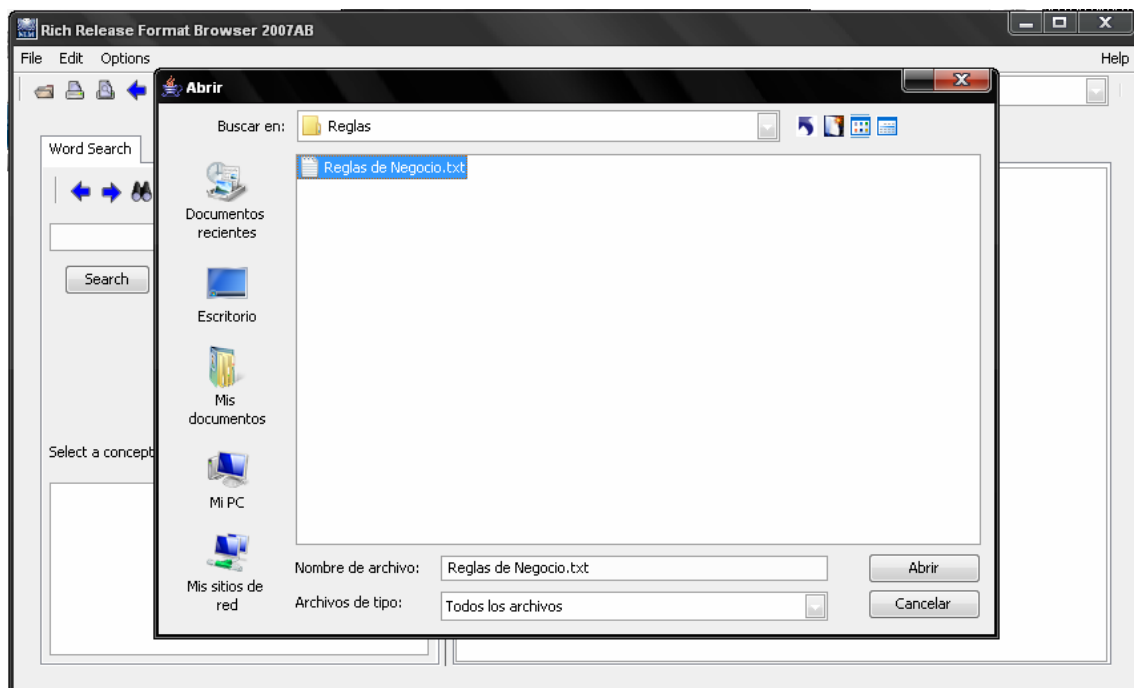


Fig. 3.21 Selección del fichero contenedor de las Reglas de Negocio.

Seleccionando el fichero correspondiente el sistema carga las reglas que estén escritas en el fichero y las almacena en una variable del sistema para mediante los controles mostrados en la figura 3.22 sean cargadas para su análisis en el propio cuadro de edición del navegador, tal y como si se estuviese escribiendo directamente en el mismo (Fig. 3.15).



Fig. 3.22 Cargar las Reglas de Negocio desde el fichero seleccionado.

Si el fichero fue creado manualmente se debe atender al formato estricto para que el funcionamiento del analizador semántico sea correcto.

Se debe asegurar la existencia del fichero localmente.

Luego de que la regla está cargada en el navegador, se procede idénticamente como explica el epígrafe anterior para la validación de una regla en el navegador.

### **3.2.3 Consulta de términos independientes.**

Esta función permite al usuario consultar o explorar los términos independientemente de que estén contenidos en una regla o se esté tratando de asegurar la validez semántica de los mismos.

El usuario escribe el término a buscar en el área de edición y mediante la selección del botón de búsqueda (Fig. 3.23) puede explorar el conjunto de definiciones y relaciones del término específico para objetivos de información o estudio.



Fig. 3.23 Botón de búsqueda de términos independientes.

### **3.3 Conclusiones parciales.**

De la implementación del sistema podemos concluir que los usuarios se verán beneficiados con el uso de una herramienta que les permitirá modificar o hasta crear Reglas de Negocio sin la necesidad de acudir a un especialista y validarlas semánticamente con rapidez. Un rasgo del sistema es la modularización, expresado en la confección de una clase que es capaz de interactuar, mediante un repositorio o fichero externo, con un editor sintáctico.



## ***Conclusiones***

La valoración, estudio y caracterización del UMLS, hace concluir que el mismo es un sistema óptimo para el trabajo con las ontologías médicas, brinda un lenguaje médico completo, con diferentes niveles de especificidad para las preguntas de consulta hechas por los usuarios. Tiene implementadas varias interfaces para la personalización de los diccionarios y vocabularios a utilizar en vista de brindar la funcionalidad suficiente para no caer en análisis de información que sea inútil o dañina para su procesamiento o consulta.

Sus servidores se ocupan de los análisis léxicos y semánticos de los términos que se le brindan para efectuar sus consultas así como brindar un trasfondo de vocabularios para satisfacer casi cualquier consulta o pregunta realizada que esté relacionada con el dominio en que se especializa la misma, lo cual implica una facilidad extraordinaria para el análisis y edición semántica de las Reglas de Negocio sea cual sea su patrón o tipo.

Seleccionar el UMLS como sistema ontológico para la validación de las Reglas de Negocio implica, para la implementación del Sistema de Validación, la selección de la tecnología del Software Libre y la especificación del lenguaje y herramienta a usarse en su implementación.

Se diseñaron e implementaron las funciones y elementos visuales para la validación de las Reglas de Negocio con las cuales se agiliza y se asegura la completa validez de las reglas que componen el proceso de nefrología.

Concluyendo con el cumplimiento de los objetivos se logra implementar el Sistema de Validación Semántica de Reglas de Negocio basándose en el UMLS.

**Recomendaciones**

## ***Recomendaciones***

Para enriquecer y perfeccionar el trabajo hasta aquí realizado, recomendamos:

- Incluir en la implementación del navegador los módulos pertinentes para el análisis sintáctico y léxico de las reglas de negocio a fin de crear un software de validación que no dependa de herramientas externas.
- Traducir al idioma español la ontología usada para la creación de este trabajo y configurarla a las necesidades específicas de las distintas áreas donde se usará el enfoque de Reglas de Negocio.
- Extender las facilidades de este sistema diseñando otras ontologías que sean compatibles con el formato del *UMLS* para su uso en otros dominios de especialización.

## **Referencias Bibliográficas**

1. UMLSKS Home Page,
2. User's and Developer's Guides.
3. ALARCON, R. (2000a) Casos de uso y actores. *Diseño Orientado a Objetos con UML*.
4. ALARCON, R. (2000b) *Diseño Orientado a Objetos con UML*.
5. DACHEL KLEINSORGE, J. W., ALLEN BROWNSE, ALAN ARONSON  
AMIA 2006 Tutorial T12 UMLS.
6. GROUP., T. B. R. (2000) Defining Business Rules – What Are They Really?
7. GRUBER, T. R. (1993) A Translation Approach to Portable Ontology  
Specification., 199-220.
8. GUARINO, N. (1998) Formal Ontologies and Information Systems. 3-15.
9. MIZOGUCHI, R. (1997) Roles of shared Ontology in Ai-ED research. 537-544
10. OLIVIER BODENREIDER, J. W., WILLIAM HOLE The Unified Medical  
Language System(2004\_medinfo\_tut.pdf). National Library of Medicine(NLM).
11. TORRES, P. L. (2002) Diagrama de Actividades.



Anexo #1 Diagrama de Actividades para el proceso de Validación Semántica de Reglas de Negocio.

