



UNIVERSIDAD CATÓLICA DEL NORTE  
FACULTAD DE INGENIERÍA Y CIENCIAS GEOLÓGICAS,  
MAGÍSTER EN INGENIERÍA INFORMÁTICA

Departamento de Ingeniería de Sistemas y Computación

GENERACIÓN AUTOMÁTICA DE EXPLICACIONES DESDE  
UNA BASE DE REGLAS Y UNA ONTOLOGÍA DE DOMINIO

Tesis para optar por el título de Magíster en Ingeniería Informática

**YAHIMA HADFEG FERNÁNDEZ.**

Profesores guías:           Dr. Víctor Flores Fonseca.  
  Dr. Claudio Meneses Villegas.

Antofagasta, 2018

## TABLA DE CONTENIDOS

<b>CAPÍTULO I. INTRODUCCIÓN.....</b>	<b>- 1 -</b>
1.1. SISTEMAS DINÁMICOS .....	- 1 -
1.2. PLANTEAMIENTO DEL PROBLEMA .....	- 2 -
1.2.1. Preguntas de Investigación.....	- 2 -
1.2.2. Hipótesis de trabajo.....	- 3 -
1.3. OBJETIVOS.....	- 3 -
1.3.1. Objetivo General.....	- 3 -
1.3.2. Objetivos específicos.....	- 3 -
1.3.3. Tareas de Investigación .....	- 4 -
1.4. ALCANCE Y CONTRIBUCIÓN DEL TRABAJO .....	- 5 -
1.5. ESTRUCTURA DEL DOCUMENTO DE TESIS.....	- 6 -
<b>CAPÍTULO II: MARCO TEÓRICO.....</b>	<b>- 7 -</b>
2.1. MODELOS DE SISTEMAS DINÁMICOS.....	- 7 -
2.2. REPRESENTACIÓN DEL CONOCIMIENTO .....	- 8 -
2.2.1. Técnicas de representación del conocimiento.....	- 11 -
2.2.1.1. Lógica de predicados .....	- 11 -
2.2.1.2. Redes semánticas .....	- 12 -
2.2.1.3. Ontologías.....	- 13 -
2.3. GENERACIÓN DE EXPLICACIONES.....	- 23 -
2.3.1. Propiedades de la Explicación .....	- 24 -
2.3.2. Explicaciones en sistemas expertos .....	- 28 -
2.3.2.1. MYCIN .....	- 28 -
2.3.2.2. NEOMYCIN .....	- 29 -
2.3.2.3. XPLAIN .....	- 29 -
<b>CAPÍTULO III: ESTADO DEL ARTE .....</b>	<b>- 31 -</b>
3.1. VISIÓN GENERAL .....	- 31 -
3.2. MÉTODOS PARA LA GENERACIÓN DE EXPLICACIONES .....	- 35 -

3.2.1. Método de Explicación Reconstructiva.....	- 35 -
3.2.2. Método de Explicación HotSpot Explanations.....	- 35 -
3.2.3. Método de Gautier y Gruber.....	- 38 -
3.2.4. Método de Iwasaki y Simon.....	- 39 -
3.2.5. Método de Molina y Sánchez-Soriano.....	- 40 -
3.3. MODELOS DE PRESENTACIÓN DE MULTIMEDIA.....	- 42 -
3.3.1. Modelo propuesto por Flores.....	- 42 -
3.3.2. Modelo de Presentación Robot-Guía.....	- 45 -
3.3.3. Modelo de Presentación Encripnauta.....	- 47 -
<b>CAPÍTULO IV: GENERACIÓN DE EXPLICACIONES EN SISTEMAS DINÁMICOS.....</b>	<b>- 49 -</b>
4.1. VISIÓN GENERAL.....	- 49 -
4.2. VISIÓN GENERAL DEL SISTEMA DINÁMICO.....	- 52 -
4.2.1. Representación del sistema dinámico.....	- 54 -
4.2.2. Representación del Sistema Experto.....	- 57 -
4.2.2.1. Descripción del evento WorkingMemoryLogger.....	- 59 -
4.3. TAREA EXTRAER.....	- 61 -
4.4. TAREA PLANIFICAR.....	- 65 -
4.4.1. Operadores de presentación.....	- 67 -
4.4.2. Patrones de Discurso.....	- 69 -
<b>CAPÍTULO V: GENERACIÓN DE EXPLICACIONES EN EL DOMINIO DE LA BIOLIXIVIACIÓN - 74</b>	<b>- 74 -</b>
-	
5.1. RESUMEN DEL CAPÍTULO.....	- 74 -
5.2. GENERALIDAD DEL TRABAJO.....	- 75 -
5.2.1. Descripción del dominio.....	- 76 -
5.2.2. Representación ontológica del dominio.....	- 78 -
5.2.2.1. Representación del dominio en el Sistema Experto.....	- 86 -
5.2.2.2. Modelo de Presentación.....	- 90 -
5.3. RESULTADOS Y DISCUSIÓN.....	- 92 -

5.3.1. Selección de los casos Típicos .....	- 93 -
5.3.2. Evaluación de la utilidad práctica.....	- 94 -
5.3.2.1. Técnica post-edición.....	- 94 -
5.3.2.2. Contraste con el experto .....	- 97 -
<b>CAPÍTULO VI: CONCLUSIONES.....</b>	<b>- 103 -</b>
6.1. RESUMEN.....	- 103 -
6.2. PRINCIPALES APORTES.....	- 104 -
6.3. LIMITACIONES Y TRABAJOS FUTUROS .....	- 106 -
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>- 108 -</b>
<b>ANEXOS .....</b>	<b>VI</b>

## ÍNDICE DE TABLAS

TABLA 1. TIPOS DE ONTOLOGÍAS .....	- 20 -
TABLA 2 PRINCIPALES PROPIEDADES DE LOS MÉTODOS DE EXPLICACIÓN .....	- 24 -
TABLA 3. RESUMEN DEL ESTADO DEL ARTE PARA LA GENERACIÓN DE EXPLICACIONES. ....	- 32 -
TABLA 4 RESUMEN DEL ESTADO DEL ARTE PARA LA PRESENTACIÓN DE INFORMACIÓN. ....	- 33 -
TABLA 5 DIRECTIVAS PARA LA GRÁFICA.....	- 68 -
TABLA 6. PATRONES DE DISCURSO .....	- 71 -
TABLA 7. OBJETIVOS COMUNICATIVOS .....	- 72 -
TABLA 8. TÉRMINOS EXTRAÍDOS DEL DOMINIO A REPRESENTAR .....	- 80 -

## ÍNDICE DE FIGURAS

FIGURA. 1 EJEMPLO DE ESTRUCTURA DE DISCURSO, PATRONES DE DISCURSO Y OPERADORES [9].....	- 44 -
FIGURA 2 ARQUITECTURA DE LA PROPUESTA DE ROBOT-GUÍA [76] .....	- 46 -
FIGURA 3. ARQUITECTURA GENERAL DE LA PROPUESTA (LAS ELIPSES REPRESENTAN TAREAS, LOS CILINDROS REPRESENTAN CONOCIMIENTO PREVIO, LOS RECTÁNGULOS REPRESENTAN DATOS).....	- 51 -
FIGURA 4. REPRESENTACIÓN DEL XML DE UNA ONTOLOGÍA DESARROLLADA EN PROTÈGÈ. ....	- 57 -
FIGURA 5. LÍNEAS DE CÓDIGO PARA OBTENER EL FILELOGGER. ....	- 58 -
FIGURA 6. FRAGMENTO DEL FICHERO WORKINGMEMORYLOGGER OBTENIDO. ....	- 59 -
FIGURA 7. PSEUDOCÓDIGO DE LA TAREA EXTRAER.....	- 62 -
FIGURA 8. PSEUDOCÓDIGO DEL MÉTODO "EXTRACCIÓN DE LOS HECHOS DEL ÁRBOL DE INFERENCIA".....	- 62 -
FIGURA 9. PSEUDOCÓDIGO DEL MÉTODO "IDENTIFICACIÓN DE HECHOS Y REGLAS ACTIVADAS". ....	- 63 -
FIGURA 10 PSEUDOCÓDIGO DEL MÉTODO "EXTRACCIÓN DE ELEMENTOS ACTIVADOS". ....	- 64 -
FIGURA 11. PSEUDOCÓDIGO DEL MÉTODO "BUSCAR ELEMENTOS REPRESENTATIVOS EN RDF/XML".....	- 65 -
FIGURA 12 EJEMPLOS DE PLANTILLAS DE FRASES .....	- 67 -
FIGURA 13. EJEMPLO DE EXPLICACIÓN GENERADA CON PLANTILLA DE FRASES.....	- 68 -
FIGURA 14. PSEUDOCÓDIGO DE LA TAREA PLANIFICAR .....	- 73 -
FIGURA 15 ESQUEMA DE UNA PILA DE BIOLIXIVIACIÓN DE COBRE. ....	- 77 -
FIGURA 16. VISTA DE PILA Y PISO .....	- 78 -
FIGURA 17. ESQUEMA DE LA PILA .....	- 78 -
FIGURA 18. JERARQUÍA DE CLASES.....	- 81 -
FIGURA 19. PROPIEDADES DE OBJETOS.....	- 81 -
FIGURA 20. PROPIEDADES DE DATOS.....	- 81 -
FIGURA 21. GRÁFICO QUE REPRESENTA LA ONTOLOGÍA.....	- 82 -
FIGURA 22. FRAGMENTO DEL RDF/XML OBTENIDO DEL MODELADO. ....	- 86 -
FIGURA 23. DIAGRAMA DE OBJETOS DEL MÓDULO DE TEMPERATURA UTILIZADO POR EL SISTEMA EXPERTO. ....	- 88 -
88 -	
FIGURA 24. FICHERO LOG OBTENIDO DEL DROOLS COMO RESULTADO DE LA INFERENCIA.....	- 90 -
FIGURA 25 ESQUEMA DE PLANIFICACIÓN DE LA PRESENTACIÓN .....	- 91 -

FIGURA 26 DISTRIBUCIÓN DE LOS DATOS .....	- 93 -
FIGURA 27. CASOS TÍPICOS IDENTIFICADOS .....	- 94 -
FIGURA 28. EJEMPLO DE TEXTO ORIGINAL CON PALABRAS MODIFICADAS.....	- 96 -
FIGURA 29. EJEMPLO DE TEXTO MODIFICADO CON PALABRAS AÑADIDAS.....	- 97 -
FIGURA 30 TABULACIÓN DE LA ENCUESTA .....	- 98 -

# Resumen

La exposición de un asunto, doctrina o texto con claridad suficiente para que se haga más perceptible es lo que conocemos por explicar. Las explicaciones de fenómenos, procesos y situaciones ayudan a los seres humanos a entender lo que sucede y tomar decisiones acertadas, de ser requeridas. Actualmente existen muchos sistemas creados por el hombre que apoyan la toma de decisiones, fundamentalmente enfocados en las empresas.

Muchos de los sistemas que apoyan la toma de decisiones son materializados a través de sistemas expertos. Estos sistemas expertos se sustentan en su base de conocimiento y mecanismo de inferencia para obtener las conclusiones. Sin embargo, no todos los sistemas expertos cuentan con los mecanismos para explicar el proceso inferencia que llevaron a cabo para dar un resultado.

El presente trabajo se ha centrado en la investigación de una solución para automatizar la tarea de generación automática de explicaciones del comportamiento de sistemas dinámicos en forma de presentaciones multimedia.

En este trabajo se han analizado soluciones del campo de la Inteligencia Artificial acerca de métodos de generación automática de explicaciones y representación de sistemas dinámicos.

Como resultado del trabajo, se ha planteado un método general basado en el conocimiento, que es aplicable a sistemas dinámicos. La solución propuesta incluye una forma de representación general del conocimiento que como mínimo abarca los conceptos del modelo de objeto utilizado para la construcción del Sistema Experto.

Para la validación de la investigación se escogió el dominio de la Biología, el cual permitió extraer conclusiones sobre la validez y utilidad práctica de la solución planteada.

# Capítulo I. Introducción

---

## 1.1. Sistemas Dinámicos

Muchos de los sistemas hoy día cuentan con estados físicos que evolucionan en el tiempo. Los estados físicos pueden ser obtenidos con sensores y llegar a ser numerosos (esto es lo que se conoce como Sistema Dinámico [1], [2]) Gestionar por un humano el volumen de información entregada por los sensores en un tiempo relativamente corto, para tomar decisiones, resulta inviable. Los sistemas expertos son herramientas que apoyan la toma de decisiones y tienen la capacidad de manejar grandes volúmenes de datos para llegar a conclusiones acertadas usando mecanismos de inferencias [3].

Para los usuarios de un Sistema Experto resulta interesante explicar las conclusiones a las que llega dicho sistema, basadas en el proceso de inferencia asociado. Con ello se puede entender con mayor claridad el resultado ofrecido y la dinámica del sistema en cuestión. Con la explicación también se puede observar los estados por los que transitó el sistema antes de llegar al final de su proceso de inferencia y los valores que tomaron las distintas variables al activar una determinada regla.

En un contexto de explotación o gestión de un sistema dinámico [1], explicar el proceso de inferencia de un Sistema Experto [4], permite al experto y usuario, encontrar posibles relaciones entre parámetros o variables que no serían fáciles de identificar manipulando gran cantidad de datos en cortos períodos de tiempo o en ciertas situaciones. Dichas relaciones pueden ser convertidas en nuevo conocimiento, el cuál puede ser adicionado a la base de reglas del Sistema Experto para obtener mejores resultados. Las explicaciones también facilitarán el entendimiento de la dinámica del sistema permitiendo corregir acciones no deseadas en el futuro.

Para poder realizar explicaciones sobre los resultados arrojados por el Sistema Experto se debe tener un conocimiento previo del dominio y este conocimiento debe ser representado. Existen diversas técnicas de Inteligencia Artificial [5] para representar información y/o conocimiento. La ontología es una de esas técnicas que mayor nivel de representatividad tiene, permitiendo una alta riqueza en la descripción de un dominio determinado.

## **1.2. Planteamiento del Problema**

Bajo las consideraciones anteriores, el *problema de investigación* se puede definir de la siguiente manera:

A pesar de las bondades que brinda la implementación de un Sistema Experto y la expresividad de las ontologías para representar el conocimiento, se requiere un método capaz de explicar automáticamente las conclusiones arrojadas por un Sistema Experto basado en su proceso de inferencia.

En este trabajo se quiere desarrollar un método de generación automática de explicaciones apoyado en una ontología de dominio, para facilitar de esta manera el entendimiento de las conclusiones arrojadas por el Sistema Experto. De esta forma se pueden mostrar las posibles relaciones que se dan dentro del sistema.

### **1.2.1. Preguntas de Investigación**

De este planteamiento surge la motivación de esta investigación, la cual parte de las siguientes preguntas iniciales:

1. ¿Cómo presentar el proceso de inferencia de un Sistema Experto lo más parecido a la forma en que lo haría un experto humano?
2. ¿Cómo explicar los resultados arrojados por un Sistema Experto en su proceso de inferencia usando ontologías de dominio y los estados por los que transitó para llegar a la conclusión?

3. ¿Cómo vincular una ontología de dominio con una base de reglas en un sistema, de forma que se obtengan descripciones entendibles por un humano del proceso de inferencia?

### **1.2.2. Hipótesis de trabajo**

En base a una ontología de un dominio y una base de conocimiento asociada a ese dominio se puede generar de forma automática una explicación cercana a la de un humano sobre el proceso de inferencia y conclusión alcanzada por un Sistema Experto.

## **1.3. Objetivos**

Habiéndose definido la hipótesis del trabajo y teniendo en cuenta la problemática presente en los sistemas expertos desarrollados en la actualidad, se plantea el objetivo general y los objetivos específicos de la presente investigación.

### **1.3.1. Objetivo General**

Diseñar un algoritmo de construcción automática de explicaciones similar a la que generaría un experto humano, utilizando una ontología de dominio y la base de reglas del Sistema Experto, para facilitar la comprensión del proceso de inferencia y la conclusión alcanzada por el Sistema Experto.

### **1.3.2. Objetivos específicos**

1. Investigar técnicas actuales de Representación del Conocimiento, Generación de Explicaciones y Presentación de Información para Sistemas Dinámicos.
2. Diseñar una ontología en el dominio seleccionado como caso de estudio.
3. Diseñar un algoritmo que combine las reglas gestionadas por un Sistema Experto y una ontología de dominio para generar explicaciones.
4. Validar los resultados obtenidos.
5. Difundir los resultados en medios científicos.

### 1.3.3. Tareas de Investigación

Por cada uno de los objetivos específicos identificados en la sección anterior se identificaron tareas de investigación. Estas tareas de investigación responden a actividades concretas que se deben realizar para alcanzar el objetivo.

**Objetivo 1:** Investigar técnicas actuales de Representación del Conocimiento, Generación de Explicaciones y Presentación de Información para Sistemas Dinámicos.

1. Desarrollar un marco teórico sobre Representación del Conocimiento para Sistemas Dinámicos.
2. Desarrollar un marco teórico sobre Generación de Explicaciones y Resúmenes de Información para Sistemas Dinámicos.
3. Desarrollar un marco teórico sobre Patrones de Presentación de información para Sistemas Dinámicos.

**Objetivo 2:** Diseñar una ontología en el dominio seleccionado como caso de estudio.

1. Seleccionar el dominio de aplicación.
2. Analizar el proceso relacionado con el dominio seleccionado.
3. Identificar los componentes del proceso seleccionado como caso de estudio y las relaciones entre ellos.
4. Poblar la ontología.

**Objetivo 3:** Diseñar un algoritmo que combine las reglas de una base de conocimiento y una ontología de dominio para generar explicaciones.

1. Analizar la herramienta Drools que da soporte a las reglas obtenidas por el Sistema Experto.
2. Analizar formas de relacionar la ontología con la base de reglas.
3. Identificar las clases y componentes necesarios para la implementación del algoritmo.

4. Seleccionar los patrones de discurso en el Modelo de Presentación.

**Objetivo 4:** Validar los resultados obtenidos.

1. Utilizar métricas existentes para comparar la distancia entre los textos generados por el algoritmo y un experto humano.
2. Utilizar la retroalimentación de un experto humano en el dominio para validar los resultados.

**Objetivo 5:** Difundir los resultados en medios científicos.

2. Publicar los resultados preliminares en congresos o revistas.
3. Publicar los resultados finales en una revista de alto impacto.

## **1.4. Alcance y Contribución del trabajo**

Aunque existen múltiples técnicas para la creación de sistemas expertos y con ellas algunas alternativas de cómo explicar las conclusiones arrojadas o caminos escogidos por éste, hay muchos espacios aún por cubrir. Uno de ellos es el estudio de la combinación de técnicas de Inteligencia Artificial como ontologías, patrones de discurso y planificación inteligente, aplicadas a la generación automática de explicaciones.

Por ello, el alcance de esta investigación pasa por la definición y construcción de un algoritmo capaz de vincular una ontología de dominio con una base de reglas para obtener una explicación del proceso de inferencia de un Sistema Experto. Dentro de este algoritmo deberá definirse una manera efectiva de recorrer la ontología y tomar de ésta lo necesario para la elaboración de la explicación. También se deberá contar con un método eficaz y no estático para presentar la información que explique la salida de un Sistema Experto.

El aporte de la investigación se puede resumir en dotar a los sistemas expertos con mecanismos más inteligentes de presentar las conclusiones alcanzadas, facilitando tareas posteriores como la toma de decisiones a los usuarios del software. Constituye un aporte también, el uso de las técnicas de Inteligencia

Artificial antes mencionadas, para la construcción de un algoritmo de generación de explicaciones, que complementa al Sistema Experto.

## **1.5. Estructura del documento de tesis**

Para facilitar la comprensión de la investigación, el resto del documento se ha estructurado de la siguiente manera:

En el capítulo II se describe el contexto del problema brindando los fundamentos teóricos correspondientes a los distintos métodos que se usaron en este trabajo.

En el capítulo III se centra en una revisión del estado del arte dentro del contexto del problema planteado. Se revisan las técnicas más representativas, relacionadas con el objetivo de esta tesis. Al final de este capítulo se presenta una discusión que se focaliza en la relación de las técnicas y tendencias de estas áreas de investigación con los objetivos de la tesis y se puntualiza sobre la adecuación de éstas a los objetivos planteados.

En el capítulo IV se detalla la propuesta de método de generación automática de explicaciones. Este capítulo está orientado a ofrecer una solución técnica acorde a los objetivos planteados.

En el capítulo V se instancia la propuesta a un dominio real y complejo, para validar su aplicabilidad y su utilidad práctica que posibilitan la generación de explicaciones del proceso de inferencia de un Sistema Experto que caracteriza un sistema dinámico. También se incluye en este capítulo la descripción del dominio de aplicación de la propuesta representada en una ontología, así como los patrones de presentación utilizados.

El capítulo VI contiene el resumen de las contribuciones alcanzadas y se exponen las limitaciones y líneas principales de trabajo futuro a desarrollar.

Finalmente, la memoria contempla las reseñas bibliográficas y un conjunto de anexos que describen casos de prueba usados en la validación de la propuesta.

# **CAPÍTULO II: MARCO TEÓRICO**

---

## **2.1. Modelos de Sistemas Dinámicos**

Un modelo de sistema puede pensarse como el proceso de abstracción o vista simplificada de parte del mundo, que se realiza con un propósito concreto [6]. En el caso de los sistemas dinámicos, se consideran elementos para identificar su evolución en el tiempo [7]. Estos modelos se pueden representar desde distintas perspectivas, como por ejemplo, usando técnicas clásicas de ecuaciones diferenciales de la física clásica, técnicas estadísticas para predicción del comportamiento, redes bayesianas dinámicas, etc. [8].

Los modelos de sistema se pueden dividir en modelos cuantitativos y cualitativos. Los primeros se caracterizan porque los valores de las variables son magnitudes continuas (números reales), mientras que en los segundos los valores de las variables son magnitudes discretas y pueden ser representadas tanto de manera simbólica como numérica [9].

En algunas clasificaciones los modelos de sistema se dividen en modelos simbólicos o analíticos. Un ejemplo de un modelo simbólico puede ser un flujograma de un sistema de manufactura. Un modelo analítico, por su parte, puede ser reducido a un conjunto de ecuaciones que al ser resueltas permiten obtener una solución que está dentro de un conjunto de soluciones [10].

Otra posible clasificación de los modelos de sistemas, consiste en agruparlos en modelos estáticos y modelos dinámicos. Los primeros ayudan a entender el funcionamiento del sistema, cuando está en reposo o inalterado, un ejemplo de esto puede ser un modelo usado para calcular la fuerza necesaria para mover un objeto desde el reposo. Los modelos dinámicos por su parte, ayudan a entender los cambios en el sistema a lo largo del tiempo, es decir cuando dicho sistema se encuentra en funcionamiento e interactuando con otros sistemas [11].

Existen básicamente dos aspectos que caracterizan y dificultan el modelado de sistemas dinámicos: su complejidad estructural y de funcionamiento o el conocimiento parcial que se disponga de ellos. Esto hace que generalmente sean difícilmente representables.

Tanto los aspectos estáticos como los dinámicos de un sistema dinámico son de interés para comprender su funcionamiento en un contexto determinado. Entre los aspectos que generalmente se consideran básicos de representar se pueden distinguir:

- Representación de la estructura: Una representación de la estructura de un sistema habitualmente contiene representación de los elementos y las relaciones existentes entre ellos. En ciertos casos, la representación de la estructura de un sistema puede ser compleja, dependiendo del detalle que se contemple. Estos detalles estructurales suelen separarse en niveles para mejorar el manejo de los contenidos. Representar un sistema dinámico como un conjunto de elementos y sus relaciones, permite realizar tareas como derivar conocimiento del comportamiento de dicho sistema o contextualizar la ocurrencia de eventos en él.
- Representación del comportamiento: La verificación de aspectos dinámicos suele estar ligada a procesos de simulación. La simulación puede entenderse como la representación en el tiempo del comportamiento de un modelo que se corresponde con un objeto o sistema físico. La simulación es útil para comprender el comportamiento de un sistema y su evolución o para validar características de un sistema (o subsistema) [12].

## **2.2. Representación del conocimiento**

Para representar el comportamiento y la estructura de un sistema dinámico, es necesario apoyarse en técnicas de representación de conocimiento. Todo problema es más sencillo de resolver si se dispone de conocimiento específico sobre él. Este conocimiento dependiente del dominio se combina con el

conocimiento general sobre cómo resolver problemas. Este conocimiento ha de permitir guiar a los mecanismos de la Inteligencia Artificial para obtener soluciones de manera más eficiente.

La representación del conocimiento es un área de la Inteligencia Artificial cuyo objetivo fundamental es representar lo conocido de una manera que facilite obtener conclusiones a partir del conocimiento que se tiene (la inferencia) [13]. Representar el conocimiento es equivalente a escribir en algún lenguaje descripciones del mundo.

La representación es *"...un conjunto de convenciones sistemáticas y semánticas que hacen posible el describir cosas"* [14].

El conocimiento puede ser de tipo declarativo, procedimental o heurístico. El conocimiento declarativo consiste en tener un saber sobre un tema o materia al que se llega por medio de la investigación y no por la experiencia personal. Es un saber público, lógico y coherente, verificable mediante reglas de comprobación. Es abstracto, conceptual y es la base de los conocimientos académicos [15].

El conocimiento procedimental implica saber cómo hacer algo, se desarrolla por medio de las destrezas y no depende del conocimiento declarativo. Abarca las competencias para saber actuar en una determinada situación [15].

El conocimiento heurístico es un tipo especial de conocimiento usado por los humanos para resolver problemas complejos. El adjetivo heurístico significa medio para descubrir. Se entiende por heurístico a un criterio, estrategia, método o truco utilizado para simplificar la solución de problemas. El conocimiento heurístico usualmente se adquiere a través de mucha experiencia [16].

Para representar el conocimiento generalmente se usa algún tipo de lógica con el objetivo de proveer una semántica formal de cómo las funciones de razonamiento se aplican a los símbolos del dominio del discurso, además provee operadores como cuantificadores, operadores modales, etc. Esto, junto a una teoría de interpretación, da significado a las frases en la lógica.

Al diseñar un sistema utilizando una técnica de representación del conocimiento se realizan elecciones a lo largo de un número de ámbitos de diseño. La característica más importante que hay que tomar en cuenta es la expresividad de la técnica de representación del conocimiento; cuanto más expresiva es, se hace más sencillo comunicar lo que se quiere expresar. Por otra parte, cuanto más expresivo es un lenguaje, más difícil es derivar inferencias automáticamente de él.

El principal problema es encontrar una técnica de representación del conocimiento y un sistema de razonamiento que le de soporte, que pueda hacer las inferencias que necesita una aplicación dentro de los límites de recursos del problema a tratar.

Una buena representación del conocimiento cubre seis características básicas:

- Cobertura: La representación del conocimiento cubre la información a lo ancho y a profundidad. Sin una cobertura amplia, la representación del conocimiento no puede determinar nada ni resolver ambigüedades.
- Comprensible por humanos: La representación del conocimiento es vista como un lenguaje natural, así que la lógica debe fluir libremente. También debe soportar la modularidad y la jerarquía de clases, además contar con primitivas simples que se combinen de forma compleja.
- Consistencia: La información debe ser consistente en cualquier sentido que se lea, por ejemplo; si alguna persona ha encendido la computadora, también puede ser interpretado, cómo que la computadora ha sido encendida. Siendo consistente, la representación del conocimiento puede eliminar conocimiento redundante o conflictivo.

- Eficiencia: La representación cumple su objetivo a la perfección, utiliza los medios disponibles de manera racional para alcanzar su meta.
- Facilidad de modificación y actualización.
- Soporte de la actividad inteligente que usa la base de conocimiento. Puede entenderse como información necesaria para que algoritmos con cierto grado de complejidad puedan arrojar información interesante de la representación.

## **2.2.1. Técnicas de representación del conocimiento**

### **2.2.1.1. Lógica de predicados**

Muchos de los sistemas de representación del conocimiento de la actualidad se basan en algún tipo de lógica formal. La lógica formal aporta un buen número de ventajas para la representación del conocimiento y su manejo, partiendo de una sintaxis y semántica bien definidas que detallan perfectamente la forma de construir sentencias y razonamientos sobre ellas.

Una proposición es una sentencia que puede considerarse falsa o verdadera. En la lógica proposicional se asignan símbolos a cada sentencia y se utilizan operadores lógicos sobre ellos para crear proposiciones más complejas. Partiendo de los símbolos y utilizando los diferentes operadores se construyen proposiciones complejas, las cuales son posibles obtener si son verdaderas o falsas operando a partir de los valores de verdad de cada uno de los símbolos iniciales, utilizando el cálculo proposicional.

La lógica de primer orden es una ampliación de la lógica proposicional a partir de dos operadores más, el cuantificador universal ( $\forall$ ) y el existencial ( $\exists$ ). Utiliza también símbolos para representar conocimiento y operadores lógicos para construir sentencias más complejas, pero a diferencia de la lógica proposicional, los símbolos pueden representar constantes, variables, predicados y funciones.

Lo que hace a la lógica interesante desde un punto de vista lingüístico es la vinculación entre lógica y significado. Además, la contribución que puede hacer la lógica a la lingüística no se limita a proporcionar descripciones precisas de los significados de las conjunciones gramaticales, la negación, las expresiones cuantificadoras, etc. Se debe hacer notar que la lógica proporciona interpretaciones semánticas de las operaciones sintácticas [17].

#### **2.2.1.2. Redes semánticas**

El término red semántica deriva de la tesis de doctorado en psicología cognitiva de Ross Quillian [18], quien primero lo introdujo como modo de expresar la organización de la memoria semántica humana, es decir, de la memoria por conceptos expresados en palabras. Una red semántica, según la definición pionera de Ross Quillian, es un grafo en el cual están presente elementos como los nodos o vértices y arcos o flechas [18].

Las primeras redes semánticas se implementaron en sistemas de traducción automática a principios de los años sesenta. Desde entonces, se han diseñado e implementado docenas de versiones diferentes. Aunque la terminología y las anotaciones varían ampliamente, los temas siguientes son comunes a la mayoría de ellos [19]:

- Los nodos en la red representan conceptos de entidades, atributos, eventos y estados.
- Los arcos en la red, usualmente son llamados relaciones conceptuales que representan relaciones entre los nodos del concepto. Las etiquetas de los arcos especifican los tipos de relación.
- Algunas relaciones conceptuales representan casos lingüísticos, tales como agente, paciente, receptor o instrumento. Otros representan conectivos espaciales, temporales, causales y lógicos. Otros especifican el papel que una entidad juega con respecto a otra. Pero la representación de los roles como relaciones o conceptos es un área de divergencia entre los diferentes sistemas.

- Los tipos de conceptos se organizan en una jerarquía de acuerdo a los niveles de generalidad, tales como: animal, carnívoro, felino, gato. Esta jerarquía se denomina a menudo una jerarquía de tipos o una jerarquía taxonómica. También se llama jerarquía de subsunción<sup>1</sup>, ya que las instancias de un tipo general como animal subsumen<sup>2</sup> las instancias de un tipo más especializado como gato.
- Las relaciones que sostienen otros conceptos de un tipo dado se heredan a través de la jerarquía por todos los subtipos. Puesto que cada animal requiere oxígeno, la propiedad de requerir oxígeno es heredada por cada carnívoro, felino y gato.

A pesar de estos temas comunes, las redes divergen en una serie de cuestiones, tales como las filosóficas de significado, métodos para representar a todos los cuantificadores y operadores de lógica, técnicas para manipular las redes y realizar inferencias y convenciones estilísticas para dibujar los nodos y arcos y etiquetar las esquinas con palabras u otros símbolos. Algunos sistemas hacen hincapié en la capacidad de afirmar proposiciones y razones con ellos, y otros ponen más énfasis en formas de definir nuevos conceptos en la jerarquía de tipos. Algunos están diseñados para representar el lenguaje natural, y otros están diseñados para aplicaciones de sistemas expertos. Algunos tienen una base formal en la lógica, mientras que otros son mucho más informales. A pesar de las diferencias, sus semejanzas son suficientes para caracterizarlas como una familia distintiva de sistemas de representación del conocimiento.

### **2.2.1.3. Ontologías**

Una ontología es una técnica para representar el conocimiento de un área determinada, en clara conexión con la recuperación de información que contiene en entornos informáticos. Ellas ofrecen un mecanismo de representación con una gran riqueza en cuanto a las relaciones entre conceptos y sus atributos. Es ésta la

---

<sup>1</sup> Adecuación de los hechos a la ley penal.

<sup>2</sup> Incluir algo como componente.

razón por la que en la actualidad es la técnica preferida para representar el conocimiento [20].

Las ontologías se han extendido a campos como la ingeniería de conocimiento, la Inteligencia Artificial y ciencia de la computación, procesamiento del lenguaje natural [21], representación del conocimiento [22], sistema de información cooperativa [23], comercio electrónico [24], bioinformática [25], diseño e integración de bases de datos [26], integración inteligente de información, recuperación de información [27] y gestión del conocimiento [28], etc.

Existen numerosas definiciones de ontología. Una de las primeras fue la ofrecida por Neches quien en su artículo *Enabling technology for knowledge sharing* [29] plantea:

*"Una ontología define los términos básicos y las relaciones que comprenden el vocabulario de un área temática, así como las reglas para combinar términos y relaciones para definir extensiones al vocabulario".*

En 1993 Gruber [30] apoyándose en la definición anterior agrega:

*"Una ontología es una especificación explícita de una conceptualización".*

En 1997 la definición dada por Gruber fue extendida por Borst [31], definiendo una ontología como:

*"Las ontologías se definen como una especificación formal de una conceptualización compartida".*

Interesa la definición dada por Studer, Benjamins y Fensel [32], por insistir en el significado exacto de cada uno de los componentes:

*"una especificación explícita y formal de una conceptualización compartida.*

**Formal:** *Se refiere al hecho de que una ontología debe ser legible por máquina.*

**Explícita:** Significa que el tipo de concepto utilizado y las restricciones en su uso se definen de forma explícita.

**Compartido:** Refleja la idea de que la ontología captura conocimiento consensuado, es decir, que no es el privilegio de un individuo, pero aceptado por un grupo.

**Conceptualización:** Se refiere a un modelo abstracto de algún fenómeno donde se identifican los conceptos pertinentes de dicho fenómeno".

Otra de las definiciones popularizadas de ontología es la ofrecida por Noy y McGuinness [33]:

*“Una ontología es una descripción explícita y formal de conceptos en un dominio de discurso (clases o conceptos), propiedades de cada concepto describiendo varias características y atributos del concepto (slots, roles o propiedades), y restricciones sobre los slots (facetas o restricciones de rol)”.*

Una ontología junto con un conjunto de individuos de clases constituye una base de conocimiento. En realidad, hay una línea muy delgada donde la ontología termina y la base de conocimiento empieza.

### **Principales componentes de una ontología**

Las ontologías no siempre tienen la misma estructura, pero existen características comunes y componentes básicos presentes en la mayoría de ellas. Incluso con propiedades diferentes, es posible identificar los tipos más significativos.

Teniendo en cuenta que hay varias técnicas para el modelado de ontologías, existen diferentes componentes en función de la técnica usada [34]. Se distinguen entre ontologías construidas con las siguientes técnicas:

- Frame y lógica de primer orden.
- Lógica descriptiva.

- Técnicas de Ingeniería de Software.
- Técnicas de base de datos.

A continuación, se detallan los componentes de las ontologías construidas con cada técnica.

#### *Ontologías con Frame y Lógica de primer orden*

**Clases:** representan los conceptos donde las clases se organizan normalmente en taxonomías en las que pueden definirse metaclases, que son clases cuyas instancias son clases.

**Relaciones:** son asociaciones entre conceptos del dominio. Generalmente se trata de relaciones binarias en las que se especifica el dominio y el rango de la relación.

**Funciones:** son un caso especial de relación, en la que el enésimo elemento de la relación es el único para los n-1 elementos precedentes, calculando una expresión en función de los n-1 elementos precedentes.

**Axioma:** se usa para representar conocimiento cierto que no puede ser representado con los otros componentes, para inferir nuevo conocimiento y validar la consistencia del elemento almacenado.

**Instancias:** representan elementos individuales en una ontología, objetos particulares de una clase.

#### *Ontologías con Lógica descriptiva*

**Conceptos:** pueden ser primitivos si se definen mediante condiciones necesarias. Son equivalente a las clases.

**Roles:** pueden ser primitivos o derivados. Describen relaciones binarias entre conceptos y también la descripción de propiedades de conceptos.

*Individuos:* representa instancia de conceptos y los valores de sus roles (propiedades).

#### *Ontología con técnicas de Ingeniería de Software.*

Las ontologías también pueden ser definidas con técnicas de Ingeniería de Software. Un ejemplo de técnica que se usa para modelar ontología es UML (Unified Modeling Language) [35], que es un lenguaje muy extendido y de fácil uso. UML es un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objeto [36].

Para poder representar las ontologías con UML es necesario enriquecer el modelo con OCL (Object Constraint Language). OCL es un lenguaje declarativo para describir reglas para aplicar a un modelo UML. Se utiliza como lenguaje de consulta, para especificar restricciones, precondiciones y post- condiciones de las operaciones o predicados que deben cumplirse [37].

*Clases:* son representadas mediante cajas en las que se especifican el nombre, los atributos y las operaciones de la clase (las operaciones no son usadas). La taxonomía de clases se especifica mediante las relaciones de generalización entre las clases.

*Relaciones:* son expresadas mediante asociaciones UML entre clases, en las que se puede especificar la cardinalidad. Si se quiere expresar una relación n-aria, no se puede hacer directamente, es necesario hacerlo con una clase intermedia.

*Axiomas:* son equivalentes a los axiomas de los Frame y la Lógica de primer orden. Se expresan mediante OCL.

#### *Ontología con técnicas de bases de datos*

En este caso se plantea el uso de diagramas de entidad relación para modelar ontologías, que pueden ser traducidos posteriormente a SQL (Structured Query Language) [38].

*Clases:* se representan con entidades. La taxonomía de las clases se puede expresar mediante las relaciones de generalización.

*Atributos:* se puede especificar los atributos y su tipo para cada entidad.

*Relaciones ad hoc:* las relaciones entre las clases de ontologías se especifican mediante relaciones entre las entidades, pudiendo especificar las cardinalidades.

*Axiomas:* pueden ser especificados mediante restricciones de integridad.

*Instancias:* se pueden obtener mediante la sentencia *insert* de SQL. Pero SQL no es apropiado para expresar restricciones formales de forma declarativa.

En concreto, en [39] se diseña e implementa una aplicación para traducir los modelos representados mediante ontologías en OWL (Ontology Web Language) a una base de datos relacional. En este enfoque, los conceptos, relaciones, axiomas e instancias son representados como filas de la base de datos diseñada. Aunque este enfoque tiene aún limitaciones por la falta de razonadores compatibles, permite el uso, la creación de nuevos elementos y la recuperación de información con independencia de la representación en OWL.

### **Criterios de diseño de ontologías**

Cuando se decide como representar algo en una ontología, se toma una decisión de diseño. En la actualidad no hay un estándar seguido para el desarrollo, pero se puede hacer en base a unos criterios objetivos para ontologías cuyo propósito es compartir conocimiento y la interoperabilidad entre programas. A continuación, se describen algunos criterios y principios útiles para el diseño de ontologías [40]:

*Claridad:* la definición debe ser objetiva e independiente del contexto social y computacional. Si es posible se debe hacer una definición completa mediante axiomas, ya sea sólo con condiciones necesarias o suficientes, o preferiblemente con condiciones necesarias y suficientes. Igualmente, debe ser documentada con lenguaje natural.

*Coherencia:* la ontología debe ser coherente, de manera que la inferencia sea consistente con las definiciones. Los axiomas deben ser lógicamente consistentes.

*Extensibilidad:* una ontología debe ser diseñada para anticiparse al uso de su vocabulario, es decir, debería poder definir nuevos términos para usos especializados basados en el vocabulario existente, sin que ello requiera la revisión de las definiciones existentes.

*Sesgo de codificación mínima:* la conceptualización debe ser especificada a un nivel de conocimiento tal que no dependa del nivel de codificación. Se tiene una codificación sesgada cuando las decisiones se toman sólo en base a la notación o implementación. El sesgo producido por la codificación debe minimizarse.

*Mínima asignación ontológica:* una ontología debe requerir la asignación ontológica mínima suficiente para soportar el conocimiento pretendido que comparten las actividades.

*Completitud:* una definición expresada en términos de condiciones necesarias y suficientes es preferible a una con una definición parcial, expresada a partir de condiciones necesarias o suficientes.

*Máxima extensibilidad:* nuevos términos más generales o especializados deben poder añadirse a la ontología, sin que ello requiera la revisión de los términos existentes.

*Principio de distinción:* la definición de conceptos disjuntos hace que no puedan tener instancias en común. Diversificación de jerarquías para aprovechar los mecanismos de herencia múltiple.

*Modularidad:* minimiza el acoplamiento entre módulos. Minimización de la semántica entre conceptos hermanos, lo que significa que los conceptos similares son agrupados y representados usando las mismas primitivas.

*Definición de conocimiento exhaustivo:* cuando se define un concepto indicando exhaustivamente los conceptos en los que se descompone.

*Estandarización de nombres:* seguir unas pautas de nombrado, de modo que conceptos similares se nombren de forma parecida y sea más fácil entender la ontología.

*Tipos de ontología*

Algunas de las propuestas definen tipos de ontologías relacionándolos con su función [41], el grado de formalidad de su vocabulario [42], su aplicación [43], la estructura [44] y el contenido de la conceptualización [45]. La Tabla 1, resume cada enfoque.

Tabla 1. Tipos de Ontologías

<b>Enfoque</b>	<b>Clasificación</b>	<b>Descripción</b>
<b>Función</b>	Ontología de Dominio	Reutilizable en el dominio desarrollada, proporciona un vocabulario sobre concepto, sus relaciones, actividades y reglas que la rigen
	Ontología de Tarea	Proporcionan un vocabulario sistematizado de términos, especificando tareas que pueden o no estar en el mismo dominio
	Ontología General	Proporciona un vocabulario que puede estar relacionado con eventos, tiempo, espacio, comportamiento, etc.
<b>Grado de Formalismo</b>	Ontología altamente informal	Representada libremente en lenguaje natural
	Ontología semi-informal	Representada en lenguaje natural de forma restringida y estructurada
	Ontología semi-formal	Representada en lenguaje artificial definido formalmente
	Ontología rigurosamente formal	Los términos se definen con una semántica formal, teoremas y demostraciones
<b>Aplicación</b>	Ontología de autoría neutral	Una aplicación es escrita en un idioma y luego se convierte en otras para utilizar los distintos sistemas, reutilizando así la información
	Ontología como especificación	Se crea una ontología para un dominio, el cuál es usado para la documentación y el mantenimiento no en el desarrollo del software
	Ontología de acceso común a la información	Cuando un vocabulario es inaccesible, una ontología toma la información entendible, proporcionando conocimiento compartiendo

		términos
<b>Estructura</b>	Ontología de alto nivel	Describe conceptos generales relacionados con todos los elementos de la ontología, que son independiente del problema del dominio
	Ontología de dominio	Describe el vocabulario relacionado con un dominio
	Ontología de tarea	Describe una tarea o una actividad
<b>Contenido</b>	Ontología terminológica	Especifica los términos que serán usados para representar el conocimiento en un dominio
	Ontología de información	Especifica la estructura de los registros de una base de datos
	Ontología de modelación del conocimiento	Especifica las conceptualizaciones de conocimiento, tiene una estructura interna rica semánticamente y son refinadas para el uso en el dominio de conocimiento
	Ontología de aplicación	Contiene definiciones necesarias para modelar el conocimiento en una solicitud
	Ontología de dominio	Expresa conceptualizaciones que son específicas para un determinado dominio de conocimiento
	Ontología genérica	Similar a la ontología de dominio, pero los conceptos que la definen se consideran genéricos y común a muchos campos
	Ontología de representación	Explica conceptualizaciones que están detrás de los formalismos de representación del conocimiento

A principios de los noventa, las ontologías eran construidas principalmente usando técnicas de modelado de Inteligencia Artificial basadas en Frame o Lógica de primer orden (Cyc y Ontolingua)<sup>3</sup>. Posteriormente han surgido otras técnicas de representación del conocimiento basadas en técnicas de lógicas descriptivas que han sido usadas para construir ontologías y nuevos lenguajes de descripción lógica como OIL, DAM+OIL o OWL [46].

Existe una gran variedad de lenguajes para desarrollar ontologías, los principales: CycL, KIF, Ontolingua, Frame Logic, las lógicas descriptivas, SHOE, XOL, RDF(S), OIL, DAML+OIL, OWL, etc.

<sup>3</sup> Lenguajes para la representación de Ontologías.

### *Problemas con las Ontologías*

En la práctica las ontologías tienen una serie de problemas [47] a lo largo de todo su ciclo de vida, ya sea en el momento del desarrollo, cuando va a reutilizarse, combinarse o emparejarse con otra. Algunos de los principales problemas son comentados a continuación:

El primer caso descrito es la sobrecarga de la relación *es-un*, que tiene como principales consecuencias la confusión del sentido de la relación, la sobre generalización, las taxonomías de roles y la confusión de la relación “*es un tipo de*” con los roles.

Las respuestas de los expertos a veces son inapropiadas, puesto que cuando se le pide una información a un experto del dominio, éste puede responder incorrectamente, bien porque no ha entendido correctamente la pregunta, o bien porque no comprende o no conoce las implicaciones que tiene su respuesta en la ontología.

Uno de los problemas para la reusabilidad de las ontologías radica en el nivel de detalle o granularidad, que suele venir determinado por el problema que se trate. Esto hace que diferentes ontologías de un mismo dominio sean incompatibles, debido a la sobre generalización o incluso a que lo que en una son conceptos en otra son propiedades.

Existen muchos tipos de relaciones de dependencia, incluyendo niveles de realidad, entre el todo, las partes y su ambiente. Esto hace que sea necesario modelar una gran cantidad de relaciones para obtener una representación precisa.

Una de las grandes aportaciones que los investigadores proponen y ven como una ventaja es el poder eliminar la información irrelevante. Pero esto se convierte en un inconveniente cuando hay muchos modelos de información y conocimiento, y cuando se tiene que tener en cuenta a todos ellos para poder comprender el mundo real totalmente.

A menudo hay que tratar con términos ambiguos que tienen varios significados. En la aplicación de las ontologías, el principal problema que se plantea es la heterogeneidad semántica. Las mismas cosas están definidas de formas diferentes. Cada sistema o agente puede tener su propio esquema semántico para el mismo universo de discurso. Esto hace que cuando quieren comunicarse deben alinear sus esquemas. En consecuencia, se busca la integración de los dominios de aplicación que permita la interoperabilidad de los sistemas [48].

### **2.3. Generación de Explicaciones**

De acuerdo a la Real Academia de la Lengua Española una explicación se define como: "Una declaración o exposición de cualquier materia doctrina o texto con palabras claras o ejemplos, para que se haga más perceptible".

Desde el punto de vista de la ciencia moderna, la formulación de leyes y teorías científicas es lo más adecuado a la hora de prever eventos naturales y controlados. De esta manera la ciencia dispone de métodos sistemáticos muy sofisticados que describen los diversos aspectos de los fenómenos naturales para poder explicarlos.

El término "explicación" desde la perspectiva científica clásica está restringida a la descripción del "por qué" de los hechos, considerándose sólo como explicación científica aquella que busca su marco y conceptos de referencia únicamente en las relaciones causales, es decir "por qué dada ciertas condiciones, antecedentes, se podría haber esperado que ocurriese el acontecimiento a explicar".

Las explicaciones en sistemas inteligentes no son nuevas, surgen precisamente en el área de investigación de los sistemas expertos, en especial de los sistemas de reglas [49], [50]. Las investigaciones en sistemas expertos se han centrado sobre todo en las explicaciones que se pueden generar y cómo se han implementado en sistemas reales [51], [52].

### 2.3.1. Propiedades de la Explicación

Uno de los aspectos más importantes de la explicación tiene que ver con lo que a continuación se explica. En cualquier Sistema Experto hay tres cuestiones básicas para la explicación:

- La evidencia disponible,
- La base de conocimientos,
- El proceso de razonamiento realizado por el sistema para obtener (o no) una conclusión.

La Tabla 2 resume las principales propiedades de los métodos de explicación.

Tabla 2 Principales Propiedades de los métodos de explicación

Categoría	Propiedades	Opciones
<b>Contenido</b>	Enfoque	Evidencia / modelo / razonamiento
	Propósito	Descripción / comprensión
	Nivel	Micro / macro
	Causalidad	Causal / no causal
<b>Comunicación</b>	Interacción usuario-sistema	menú / preguntas predefinidas / diálogo natural / idioma
	Visualización de explicaciones	texto / gráfica / multimedia
<b>Adaptación</b>	Conocimiento del usuario sobre el dominio	hay un modelo / escala / modelo dinámico
	Conocimiento del usuario sobre el método de razonamiento	hay un modelo / escala / modelo dinámico
	Nivel de detalle	fijo/umbral/auto

#### **Contenido:** Enfoque

La explicación de las *evidencias* consiste en determinar qué hipótesis justifica los resultados observados. Por ejemplo, en los sistemas expertos sobre el proceso de Biolixiviación, tal explicación podría consistir en determinar la causa de la poca obtención de cobre en determinadas zonas de la pila, medición de la población de determinados microorganismos, etc.

La explicación del *modelo* consiste en la visualización de la información contenida en la base de conocimientos con el objetivo de ayudar a los expertos humanos en la construcción y depuración del Sistema Experto o de ofrecer al usuario mayor conocimiento sobre el dominio para fines de instrucción.

La explicación del *razonamiento* puede justificar los resultados obtenidos por el sistema y el proceso de razonamiento que los produjo. Durante la evaluación del Sistema Experto, la explicación del proceso de razonamiento es una herramienta invaluable para aislar y corregir piezas imprecisas de la información en la base de conocimientos. Por otro lado, también puede ser de interés, especialmente para los sistemas educativos, explicar los resultados obtenidos por el sistema; en particular, las conclusiones que se oponen a esta conclusión y los resultados que serían necesarios para apoyarla. Por último, cualquier herramienta de explicación también debe incluir la capacidad de realizar un razonamiento hipotético, a fin de determinar lo que se pudiera obtener como resultado del sistema si una o más de las variables observadas tomaran valores diferentes.

**Contenido:** Propósito

Existen dos objetivos diferentes (descripción / comprensión) y esto da lugar a dos tipos de explicaciones.

*Descripción.* Este tipo de explicación consiste en mostrar la base de conocimientos subyacente, o dar más detalles sobre las conclusiones, o mostrar los resultados intermedios.

*Comprensión.* En este caso, la explicación trata de que el usuario comprenda las implicaciones del modelo o las conclusiones del sistema y/o la relación entre ellos. Por ejemplo, cómo cada hallazgo afecta a la conclusión, ya sea individualmente o en combinación con otros hallazgos.

**Contenido:** Nivel

En [53] se propuso otra clasificación de los métodos de explicación para redes bayesianas, que también se pueden aplicar a otros tipos de sistemas expertos.

*Micro nivel.* Una explicación de nivel micro consiste en una justificación detallada de las variaciones que se producen en una determinada variable (o hipótesis) como consecuencia de las variaciones en sus vecinos. En el caso de un Sistema Experto basado en reglas, el nivel micro consistiría en el análisis de las variables contenidas en una determinada regla o de las normas que contienen una determinada variable.

*Nivel macro.* Este nivel de análisis de la explicación de las principales líneas de razonamiento conduce a la evidencia de una cierta conclusión. Una explicación a nivel macro para un Sistema Experto tradicional consistiría en una o varias cadenas de reglas.

### **Contenido:** Causalidad

Hay varias razones para el uso de los modelos causales en la Inteligencia Artificial. En primer lugar, los seres humanos tienden a interpretar los acontecimientos en términos de relaciones causa-efecto [54], [55]. Por lo tanto, los modelos causales son más fáciles de construir y modificar, y también los usuarios los comprenderán más fácilmente. En segundo lugar, la identificación de relaciones causales invariantes en un dominio permite la predicción de los efectos de ambas causas espontáneas (por lo general corresponden a variables aleatorias) y acciones (a veces llamados manipulaciones o intervenciones) [56]. En tercer lugar, el concepto de explicación está muy estrechamente ligado a la noción de causalidad; una de las modalidades de la explicación científica consiste en encontrar las causas de los hechos observados. Por lo tanto, algunos métodos de explicación para los sistemas expertos se han diseñado específicamente para los modelos causales, mientras que otros métodos son generales, en el sentido de que no asumen una interpretación causal del modelo.

La segunda categoría es la **comunicación**. Esto se relaciona con la forma en que la explicación va a ser presentada al usuario. Implica la interacción entre el usuario y el sistema. En algunos softwares, los usuarios pueden solicitar explicaciones al seleccionar algunas variables o las opciones de un determinado menú, en otros softwares los usuarios pueden plantear preguntas. Por otra parte, algunos sistemas sólo permiten al usuario pedir una explicación después de que el programa ha presentado sus conclusiones, mientras que otros sistemas permiten al usuario interrumpir el proceso de inferencia y solicitar una explicación. La pantalla de la explicación puede hacerse verbalmente, gráficamente, o en un entorno de hipertexto y multimedia que combina explicaciones interactivas con imágenes, vídeo y sonido.

La tercera y última de las categorías es **adaptación**. Es muy importante tener en cuenta lo que la explicación ofrece con el fin de cubrir las expectativas de los usuarios. Con este objetivo, el sistema debe tener en cuenta el conocimiento que el usuario tiene sobre el dominio modelado y el método de razonamiento realizado. La mayoría de los métodos de explicación no tienen en cuenta la variabilidad de conocimientos entre los diferentes usuarios y las explicaciones son generadas por un usuario hipotético.

Otros métodos se basan en un modelo de usuario estático, es decir, que no tienen en cuenta que el conocimiento del usuario aumenta a medida que interactúa con el sistema. En muchos casos, el modelo de usuario a menudo reduce la utilización de una escala con dos o tres categorías ("principiante, avanzado, experto"). En teoría, un Sistema Experto debe tener la capacidad de clasificar a los usuarios de forma automática, pero en la práctica, el usuario generalmente tiene que clasificarse a sí mismo.

También hay algunos métodos que poseen un modelo dinámico para cada usuario, lo que les permite adaptar la explicación al conocimiento que el usuario tiene en cada momento.

### **2.3.2. Explicaciones en sistemas expertos**

Los sistemas expertos emulan el razonamiento de un humano experto en un dominio concreto, y en ocasiones son usados por éstos. Con los sistemas expertos se busca una mejor calidad y rapidez en las respuestas, dando así lugar a una mejora de la productividad del propio experto al usar este tipo de sistemas informáticos.

#### **2.3.2.1. MYCIN**

Los primeros sistemas con explicaciones surgen con el sistema de reglas MYCIN que se trataba de un sistema para detectar y recomendar tratamientos en enfermedades infecciosas [57]. En este sistema el usuario podía hacerle preguntas de cómo había llegado a una determinada conclusión y el sistema le respondía con la traza del razonamiento que había usado a modo de explicación.

Todas estas explicaciones se realizaban de un modo textual, utilizando un lenguaje natural y números para expresar el grado de certeza, esto ofrecería al usuario un grado de transparencia en cómo el sistema había llegado a sus conclusiones. El usuario también puede elegir una explicación de por qué el sistema le había hecho una pregunta determinada.

Pronto se encontró que esta capacidad tenía defectos importantes, en primer lugar, el sistema no estaba seguro si los usuarios entendían el proceso de encadenamiento de reglas, también en ocasiones es complicado conseguir un argumento coherente cuando el número de reglas que encadena es alto. Por último, era insuficiente para responder a muchas de las peticiones de explicaciones de los usuarios. Por ejemplo, la estrategia de resolución de problemas de un Sistema Experto basado en reglas se define implícitamente en el sistema, pero no fue codificado explícitamente de forma que sea accesible o fácil de explicar a un usuario final.

### **2.3.2.2. NEOMYCIN**

NEOMYCIN [49] amplió las capacidades de MYCIN en este aspecto, fue desarrollado con el objetivo de explicar los resultado de MYCIN. Contenía un conocimiento explícito casual, representado por meta-reglas y una taxonomía de las enfermedades, de forma que el objetivo no es sólo explicar la cadena de razonamientos, sino también la estrategia de razonamiento. Por lo tanto, la finalidad es la comprensión tanto del modelo como del razonamiento. Las explicaciones son presentadas mediante texto, en lenguaje natural.

### **2.3.2.3. XPLAIN**

Otra extensión de MYCIN fue XPLAIN [50], la idea principal está basada en usar un programa automático para crear el Sistema Experto y la explicación de su razonamiento por el refinamiento de objetivos abstractos. XPLAIN tenía los siguientes elementos:

- El dominio de modelo, que contiene los hechos descriptivos del dominio y su representación mediante una red.
- Los principios del dominio del razonamiento, definidos por varios métodos y heurísticas.
- El programa automático para la generación del programa experto con el conocimiento proporcionado en los dos tipos de dominios anteriores.
- La estructura necesaria para generar la explicación, representada mediante un árbol, que se pueda desarrollar mediante la construcción del Sistema Experto.
- El traductor para expresar en lenguaje natural las explicaciones, toma como entrada el árbol generado anteriormente. Tiene dos elementos, un generador de frases, para construir frases

directamente de la base de conocimiento y un generador de respuesta, que determina qué partes de la estructura deben ser incluidos en la explicación y el nivel de detalle, teniendo en cuenta las necesidades de los usuarios y el estado del sistema.

El objetivo de XPLAIN es la comprensión del modelo y del razonamiento, mediante explicaciones generadas que analizan las principales líneas de razonamiento presentadas en un lenguaje natural. Por otra parte, el nivel de detalle se modifica durante la ejecución del sistema, dependiendo de las necesidades del usuario.

## **CAPÍTULO III: ESTADO DEL ARTE**

---

En este capítulo se presentan y discuten los trabajos relacionados con esta investigación, se describen los métodos encontrados con mayor relevancia y que sirvieron como base para el método propuesto.

Este capítulo cuenta con dos tablas (Tabla 3 y Tabla 4) las cuales resumen los principales trabajos relacionados con la generación de explicaciones y los métodos de presentación respectivamente. Luego, se analizan con mayor profundidad los métodos más relevantes de ambas áreas de estudio.

### **3.1. Visión General**

Como se mencionó anteriormente la Tabla 3 y la Tabla 4 muestran los resultados que han tenido distintas investigaciones en los últimos años para la tarea de generación de explicaciones y presentaciones multimedia, respectivamente. En secciones posteriores se detallan características de dichos métodos profundizando en los aspectos de interés en el contexto de este trabajo de tesis.

Ambas tablas están compuestas por cuatro columnas. La primera de ella refleja el año de la publicación de la investigación y está organizada ascendentemente quedando en las primeras filas los artículos publicados más recientemente. La segunda columna enmarca la investigación en el dominio donde fue probada. La tercera columna plasma un resumen del trabajo realizado mencionando las distintas técnicas utilizadas, por último, se muestran el nombre de los autores de los trabajos.

Tabla 3. Resumen del estado del arte para la generación de explicaciones.

Año	Dominio	Resumen	Autor
2017	Sistemas de recomendación	Predice cómo un usuario podría escribir sobre un elemento, basado en las calificaciones de los usuarios de diferentes elementos (redes neuronales).	Costa, Felipe; Ouyang, Sixun, <i>et al</i> [58].
	Sistemas de recomendación	Enfoque híbrido que integra de manera significativa las funciones de filtrado colaborativo y de análisis de sentimientos en una clasificación clásica basada en múltiples atributos.	Karacapilidis, Nikos; Malefaki, Sonia, <i>et al</i> [59].
2016	Malware	Explicación automática de comportamientos no deseados en malware móvil. Se combina el aprendizaje automático y técnicas de minería de texto para producir explicaciones en lenguaje natural.	Chen, Wei; Aspinall, David, <i>et al</i> , [60]
2015	Sistemas de recomendación	El objetivo del trabajo responde a hacer el proceso de recomendación más transparente y persuasivo. El marco propuesto construye explicaciones siguiendo un enfoque híbrido que incorpora el filtrado colaborativo y las características de análisis de sentimientos en el ranking clásico basado en múltiples atributos.	Charissiadis, Andreas; Karacapilidis, Nikos [61]
	Sistemas geográficos	Basándose en métodos computacionales que obtienen datos de las redes de sensores y se generan automáticamente descripciones textuales en lenguaje natural que ayudan a los usuarios a entender el significado de los datos con interpretaciones y explicaciones	Molina, Martin; Sanchez-Soriano, Javier, <i>et al</i> [62]
2014	Sistemas de recomendación	Trabajo que propone una metodología para construir textos explicativos que dependen de la técnica de recomendación como válida.	Galán, José [52]
2013	Sistemas de recomendación	Fueron utilizadas explicaciones para mostrar por qué se proporcionó una recomendación específica. Ha sido aprobada en varios campos, tales como sistemas expertos, sistemas de apoyo a las decisiones, tutoría inteligente sistemas y sistemas de explicación de datos.	Bobadilla, Jesús; Ortega, Francisco, <i>et al</i> [63]
	Traducción de gráficos en una descripción textual	Método para traducir un flujo de trabajo gráfico en una descripción textual, se analiza una estructura del flujo de trabajo gráfico detectando bloques gráficos del flujo de trabajo gráfico.	Solimano, Marco [64]
2012	Sistemas de recomendación	Enfocado en la explicación del porqué de una recomendación. Se utiliza el modelo HotSpot Explanations.	Merino de la Fuente, Alberto [51]

Aproximadamente el 55% de los trabajos recientes relacionados con la generación de explicaciones está orientado a los a los sistemas de recomendación. Esto se

debe a que apoyar la toma de decisiones en un mundo con tanta diversidad y volumen de datos es una tarea prioritaria en la actualidad [65], [66]. Estos sistemas de recomendación se enmarcan en varios subdominios como pueden ser médicos, financieros, deportivos, etc.

Dentro de las principales herramientas utilizadas para la generación de explicaciones se encuentran las redes neuronales, ontologías, procesamiento del lenguaje natural, aprendizaje automático, etc. En las próximas sesiones se explica en detalle el funcionamiento de alguno de estos trabajos. Los trabajos explicados en extensión son los que se consideraron como inspiración para esta investigación.

Al generar una explicación de un suceso o fenómeno, ésta debe ser presentada al usuario final de manera correcta para facilitar la recepción del mensaje que se desea transmitir. Se han realizado varias investigaciones de cómo ordenar visualmente esta información para que llegue de buena forma al receptor. Además, como vincularla con medios audiovisuales facilitando su entendimiento. La Tabla 4 presenta un resumen de las principales investigaciones relacionadas con la presentación de información, presenta una estructura igual que la Tabla 3.

Tabla 4 Resumen del estado del arte para la presentación de información.

Año	Dominio	Resumen	Autor
2017	Descripción de espacios 3D	Se patenta un sistema y un método para renderizar <sup>4</sup> contenido relacionado con medios y servicios web. Este sistema presenta contenidos con metadatos basados en el tiempo.	Douglas, Clarke; Michael, Dalrymple <i>et al</i> [69]
2016	Búsqueda personalizada en la Web	Se realiza una presentación de temas de interés para el usuario basada en patrones. Es utilizado un modelo de espacio vectorial con un peso en cada palabra. El peso es asignado teniendo en cuenta el patrón y la frecuencia de aparición en el mismo.	Leema, Mathew; Arun, Elias <i>et al</i> [70]
	Visualización de la información	La investigación utiliza un marco que describe y vincula elementos visuales esenciales de las plataformas de inteligencia de negocios (BI) a las dimensiones conocidas	Bačić, Dinko; Fadlalla, Adam

<sup>4</sup> El proceso de **renderizado** se desarrolla con el fin de generar en un espacio 3D formado por estructuras poligonales: una simulación realista del comportamiento tanto de luces, texturas y materiales (agua, madera, metal, plástico, tejidos, etcétera) como también de los comportamientos físicos (animación) [105].

	comercial	del coeficiente de inteligencia visual (IQ).	[71]
	Aplicado a cualquier dominio	Se propone un sistema para realizar presentaciones con comunicación bidireccional. Es empleado un enfoque híbrido de una presentación tradicional y una aplicación Web para realizar el sistema interactivo de presentación. El público puede proporcionar retroalimentación directa en diapositivas entregadas utilizando las aplicaciones web; la retroalimentación se sincroniza con las diapositivas del presentador.	Niwa, Yusuke; Shiramatsu, Shun <i>et al</i> [72]
	Sistemas de transporte carreteras	Son presentadas estrategias para tomar decisiones sobre qué información se muestra y en qué resolución. Esta información es mostrada en los dispositivos inteligentes de cada usuario. La tecnología utilizada para el desarrollo de esta propuesta fueron los agentes de software.	Costache, Sorina; Viswanathan, Vaisagh <i>et al</i> [73]
<b>2015</b>	Análisis visual de datos geoespaciales complejos	Se exhibe un método que permite presentaciones dinámicamente cambiantes de múltiples vistas. Se incorporan vistas desde diferentes fuentes y automáticamente el diseño de estas vistas en un entorno de múltiples pantallas.	Eichner, Christian; Nocke, Thomas <i>et al</i> [74]
	Evaluated en el área empresarial	En este trabajo son generalizados los modelos de presentaciones típicos para proporcionar soporte a nivel de paradigma y lograr flexibilidad en las presentaciones e identificar los requisitos clave para el desarrollo de una presentación lo suficientemente flexible a nivel de sistema	Bai, Xiaoyan; White, David <i>et al</i> [75]
<b>2014</b>	Educación	Patente que proporciona un sistema informatizado de presentación educativa. El sistema puede incluir un programa de aplicación de presentación. El programa de aplicación de presentación puede incluir un módulo de presentación.	Beavers, Jay C; Heiny, Lora <i>et al</i> [76]
<b>2013</b>	Aplicado a cualquier dominio	La propuesta se basa en la utilización de un Robot. Utiliza un enfoque específico (COBRA) como herramienta de integración. Cuenta con varios patrones para la presentación de la información y es capaz de generar nuevos	Rainer, Javier; Gómez, Jaime <i>et al</i> [77]
<b>2012</b>	Sistemas dinámicos (sistema hidrológico)	Presentaciones multimedia de resúmenes de información en sistemas dinámicos (estilo periodístico)	Molina, Martín; Flores, Víctor [78]

Como se puede observar en la Tabla 4 el dominio de aplicación en la presentación de información es muy amplio, los métodos utilizados para llevar en cada uno de los trabajos tienen varias aristas. Por el contrario de la generación de explicaciones, la presentación de información se ve en problemas más puntuales, son modelos creados para casos específicos.

## **3.2. Métodos para la Generación de Explicaciones**

### **3.2.1. Método de Explicación Reconstructiva**

Este método considera el proceso de explicación y razonamiento como dos tareas independientes con el fin de que las explicaciones se centren en la solución, su argumento es que las personas en sus explicaciones no describen todos los pasos para llegar a la solución [79]. Por lo tanto, consideran que los métodos para la generación de explicaciones deben tomar como entrada el proceso de razonamiento realizado por el sistema para obtener la solución, teniendo en cuenta el dominio de la explicación.

Todas estas ideas se ponen en práctica en el prototipo REX F [79], la idea del sistema es la siguiente: el Sistema Experto produce una serie de pistas sobre el proceso de razonamiento que son filtradas para determinar qué partes del razonamiento, se pasarán al módulo de generación de explicaciones, teniendo en cuenta las restricciones del problema. Este módulo usa el dominio de explicación para obtener los conceptos de la explicación que justifiquen las pistas de razonamiento y luego otro módulo que las traduce a un lenguaje natural.

Como se ha podido apreciar la principal característica de éste método y los sistemas expertos analizados, es que el sistema no tiene la capacidad de poder actualizar su conocimiento sobre el usuario durante la ejecución. Esto implica que el nivel de detalle de las explicaciones es fijo. En algunos de los más recientes el usuario puede variar el nivel de detalle, pero en general sobre un conjunto pequeño de valores. La interacción entre el sistema y el usuario se realiza a través de preguntas predefinidas.

### **3.2.2. Método de Explicación *HotSpot Explanations***

Este método está orientado a la creación de explicaciones cuyos objetivos principales sean los de aumentar la transparencia, validez y confianza en los sistema de recomendación [63], [80]. La propuesta [51] tratará de reconstruir la

línea de razonamiento seguida por el sistema al realizar el proceso de recomendación, consiguiéndolo a través de la introspección del mismo.

Otro aspecto importante del modelo de explicación propuesto, es que fue construido con retrospectiva, esto quiere decir, primero se ha generado la recomendación cuando su resultado ya se conoce y después en función de si el usuario del sistema lo desea puede pedir una explicación, si quiere, entonces se le reconstruye la explicación en función de los datos que haya recogido en el proceso de razonamiento. Este tipo de planteamiento permite desacoplar el proceso de razonamiento del de generación de la explicación.

El modelo propuesto considera atender las necesidades de explicación de un ingeniero del conocimiento como para un usuario final y las interfaces que le permiten generar son tanto textuales como gráficas.

El modelo propuesto está dividido en dos fases, una primera fase de diseño del sistema, y la segunda fase se refiere a la ejecución del sistema que se haya desarrollado en la fase de diseño.

La fase de diseño del modelo tiene las siguientes actividades:

- Creación. En esta primera fase es creado el modelo de dominio necesario para poder generar la explicación basándose en el funcionamiento del sistema de recomendación, por lo tanto, este modelo será una descripción del sistema de recomendación.
- Introspección. Con el modelo generado en el paso anterior, se tienen todos los conceptos y relaciones definidas. Ahora, para cada uno de esos datos se tiene que identificar en qué punto del sistema de recomendación implementado se extraen. Una vez identificados son generadas las trazas asociadas a ese punto del programa. A su vez cada una de estas trazas se asociarán a los conceptos y relaciones definidos en el modelo del sistema de recomendación y se generará un modelo del sistema de recomendación extendido, que incluirá

como antes la descripción del funcionamiento del sistema, pero ahora tendrá también cada uno de los datos identificados con la traza que identifica en qué punto se extrae.

- **Explicación.** En esta actividad se generan los modelos relacionados con las explicaciones, estos será una descripción de cómo serán las explicaciones usando el conocimiento del modelo generado en el paso anterior.
- **Implementación.** En esta etapa se genera el programa que permita realizar las explicaciones, se usan los modelos generados en las etapas anteriores y se implementarán para poder obtener el programa asociado. Las partes que tienen un mayor coste son las referentes a la adaptación de las trazas a los diferentes conceptos, sobre todo porque muchas veces estas trazas tendrán un proceso de parseo del mensaje de la traza. El otro proceso que tendrá un mayor coste será el de generación de la explicación sobre todo la parte de definir el proceso ya que no es definido en el modelo de conocimiento.

El sistema *HotSpotExplanations* tiene como entradas las siguientes:

- **Trazas:** son los datos que identificamos en la actividad de introspección de la fase de diseño. Cuando el usuario pide una recomendación este le responderá con los ítems recomendados, pero también internamente generará un XML con todas las trazas.
- **Ítem recomendado:** el usuario pedirá una recomendación al sistema de recomendación y éste le responderá con los ítem y trazas.
- **Usuario:** el usuario pide una recomendación y el sistema le responde con la recomendación, después el usuario puede pedir una explicación sobre un ítem en particular. En este punto es importante el usuario porque dependiendo del tipo que sea, ingeniero del

conocimiento o usuario final, podrá exigir una explicación de tipo ¿Por qué? o ¿Cómo?, con lo cual tanto la especificación del conocimiento como las posteriores explicaciones cambiarán mucho.

- *Especificación del conocimiento*: esta especificación será una instancia del modelo de sistema de recomendación, será un modelo que sólo contendrá los datos relacionados con el ítem recomendado.
- *Explicación*: esta será una de las instancias de los modelos de explicación definidas anteriormente.

Existen dos procesos principales que se encargaran de reconstruir la línea de razonamiento del sistema de recomendación y de generar una explicación asociada:

- Adaptación, este proceso será el encargado de realizar la adaptación de las trazas que recibe como parámetro de entrada *HotSpotExplanations* a la especificación del conocimiento en función del ítem recomendado y del tipo de usuario.
- Generación de la explicación, este proceso será el encargado de generar a partir de la especificación del conocimiento uno de los modelos de explicación que se especifiquen en la fase de diseño. Esta explicación dependerá del tipo de usuario y también del nivel de detalle que defina. Al final el usuario puede volver a pedir una explicación con un mayor nivel de detalle sobre la explicación que se le haya mostrado.

### **3.2.3. Método de Gautier y Gruber**

La propuesta de Gautier y Gruber consiste de un método para generar explicaciones en lenguaje natural del comportamiento obtenido por simulación de dispositivos físicos que componen sistemas dinámicos en áreas técnicas como la termodinámica o la electromecánica [81]. La propuesta se inspira en el hecho de

que, en ingeniería, el comportamiento de los dispositivos físicos se modela normalmente mediante ecuaciones matemáticas, en las que se usan variables que adquieren valores continuos en el tiempo. La instanciación de estas variables en procesos de simulación suele generar abundante información y de difícil interpretación, por lo que, un sistema que genera explicaciones en lenguaje natural de este tipo facilita el entendimiento del comportamiento del sistema.

En este trabajo la explicación del comportamiento corresponde a una presentación que contiene secuencias de explicaciones de comportamiento que se encadenan causalmente, acerca del funcionamiento de los dispositivos físicos que componen un sistema dinámico. Es decir, se expresan los estados que experimentan dichas piezas y las relaciones causales entre estos estados.

Las desventajas de este método consisten en que la información mostrada es particular a una única posibilidad de funcionamiento, ya que el esquema de construcción de la presentación consiste en ir profundizando en detalles explicativos, sin considerar otras posibles explicaciones al mismo nivel de detalle. Otra limitante de este trabajo es que las restricciones usadas para determinar estados de cada pieza del modelo están diseñadas para actuar sobre los valores de las variables que caracterizan a cada una de estas piezas, lo que le hace particular a un modelo de sistema específico y, por tanto, no reutilizable en dominios diferentes.

#### **3.2.4. Método de Iwasaki y Simon**

El método propuesto por Iwasaki y Simon tiene como objetivo determinar el comportamiento en un sistema dinámico centrado en los cambios de estados de los componentes. Esta propuesta [82] puede considerarse como una continuación del trabajo realizado por de Kleer y Brown. Básicamente, Kleer y Brown proponen que el comportamiento de un sistema se puede obtener mediante composición o agregación de los comportamientos individuales de sus componentes.

La propuesta de Iwasaki y Simon aporta a la anterior, un método para establecer comportamiento causal entre componentes y agregación de relaciones causales entre componentes. La idea del comportamiento causal (causalidad) está basada en la observación de una cadena de efectos ocasionados por la propagación de una alteración del estado inicial de un componente del sistema. La causalidad es modelada mediante relaciones de dependencia causal, establecidas entre variables (cantidades) de componentes. Para una relación causal establecida entre variables de un modelo, es posible establecer otra simétrica. Por ejemplo, si **X** causa **Y**, también puede establecerse que **Y** es-causado-por **X**.

Respecto a la agregación causal, argumentan que, en ciertos sistemas físicos con cantidades significativas de componentes, puede ser conveniente abstraer a estados más generales sobre los que sea más fácil razonar respecto a la causalidad. El método de agregación de causalidad de estados básicamente consiste en agrupar los componentes (según variables que les caracterizan) a componentes de más altos niveles (o subsistemas) y redefinir relaciones causales entre estos nuevos elementos, cuidando que estas nuevas relaciones sean compatibles con las iniciales.

Ambas ideas se basan en la capacidad intuitiva que tienen las personas, para generalizar y explicar ciertas situaciones. Una desventaja notoria de este método es que la agregación de los procesos causales depende de la estructura lineal del modelo causal, mientras que en el mundo real, esta linealidad no existe. Otra limitante importante en esta propuesta es que el conocimiento que puede expresarse en las relaciones causales, generalmente no es de carácter global. Sin embargo, esta forma de establecer un orden causal en sistemas dinámicos representa una estrategia válida en muchos casos para razonar y explicar el comportamiento de un sistema complejo.

### **3.2.5. Método de Molina y Sánchez-Soriano**

Este método describe cómo pueden ser generadas descripciones textuales de forma automática utilizando los datos recogidos por redes de sensores para la

gestión de los sistemas dinámicos, con un enfoque en las referencias geográficas. Este método resuelve el problema conocido como generación más general de la expresión [83] como es reconocido por la comunidad dedicada al estudio de la generación de lenguaje natural.

Los autores del método presentados en [62] utilizan el conocimiento geográfico detallado por ejemplo: coordenadas, lugares, distancias, formaciones naturales, *etc.*, que no siempre es fácil de adquirir y mantener al día, utilizando la disponibilidad de información geográfica en la Web por ejemplo: OpenStreetMap, Geonames, *etc.*

La idea básica es utilizar métodos computacionales que obtienen datos de las redes de sensores y generar automáticamente descripciones textuales en lenguaje natural que ayudan a los usuarios a entender el significado de los datos con interpretaciones y explicaciones.

Un ejemplo de la solución de los datos a texto se implementó para el sistema VSAIH (*Vigilancia Hidrológica Automática - Sistema de Información SAIH*) [84]. Este sistema analiza los datos de la red de sensores nacional hidrológico SAIH y sigue un estilo periodístico para generar noticias para el público en general. VSAIH genera automáticamente las noticias relacionadas principalmente con análisis descriptivos (*lo que pasó*) y también de análisis de diagnóstico (*¿por qué ocurrió?*) sobre el riesgo de inundaciones, la disponibilidad de los recursos hídricos, y fallos del sensor (ver [10] para más detalles).

El método además de generar descripciones en lenguaje natural para redes de sensores también puede proporcionar más flexibilidad en combinación con convertidores de texto a voz para utilizar en contextos de comunicación multimodal (por ejemplo, comunicación de voz, con las redes de sensores o informar a las personas con discapacidad visual con mensajes de audio sobre los datos del sensor, *etc.*).

### **3.3. Modelos de Presentación de Multimedia**

Un modelo de presentación reúne los criterios que permiten construir automáticamente una presentación que muestra de forma resumida información correspondiente al comportamiento de un sistema. Para describir el comportamiento de sistemas complejos es útil combinar formas de presentación como las siguientes:

- Texto en lenguaje natural, para expresar de una manera uniforme los contenidos principales de los resúmenes. Esta forma de expresión resulta muy conveniente, además, para ser difundida a través de diversos medios tales como Internet (por ejemplo, con mensajes de correo electrónico), móviles (con mensajes SMS o mensajes de voz), etc.
- Elementos gráficos para un mejor entendimiento de la estructura y comportamiento de los elementos del sistema. Por ejemplo, gráficos con evoluciones temporales para expresar de forma detallada los valores numéricos de variables asociadas al sistema, vídeos, fotografías, etc.

En la presentación de la información se persigue establecer una comunicación afín y fiable para el usuario, habituado a recibir este tipo de información. Es decir, se trata de usar formas de presentación habituales, que permitan vincular las diferentes piezas de información, y que el conjunto resulte cercano y convincente para el usuario.

A continuación, se exponen los modelos más significativos encontrados en la literatura.

#### **3.3.1. Modelo propuesto por Flores**

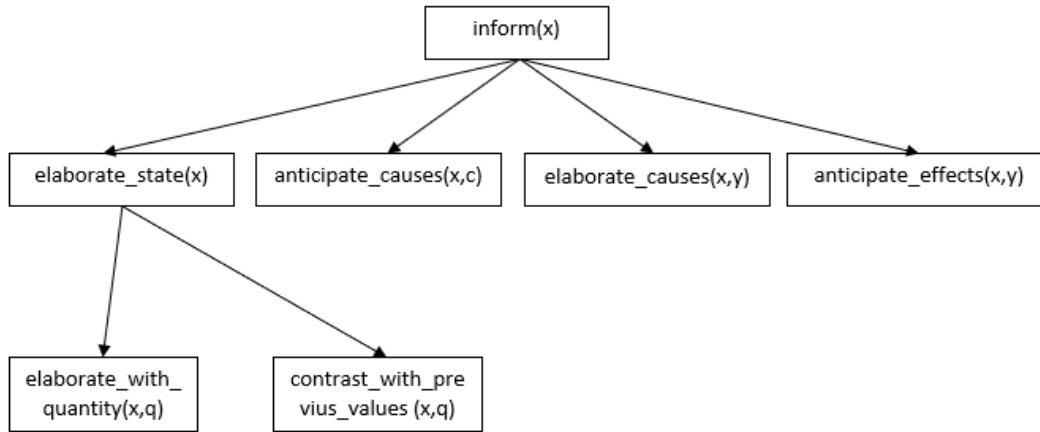
El modelo propuesto por Flores [10] para las presentaciones multimedia, está orientado a presentar resúmenes de información en sistemas dinámicos. Esta

propuesta está diseñada para ser usada en la monitorización del comportamiento de sistemas dinámicos medidos por redes de sensores. El dominio donde fue probada la propuesta fue el hidrológico, constatando su utilidad operativa mediante una construcción que recibe datos de sensores y genera resúmenes de comportamiento mostrado mediante presentaciones multimedia de cuencas hidrográficas de España en tiempo real.

El modo de abordar la generación dinámica de presentaciones, para lograr transmitir de la mejor manera la información a los destinatarios en forma de sumarios es con un planificador. En esta propuesta fue usado un planificador jerárquico (una simplificación del planificador HTN (Hierarchical Task Network) para construir el plan de presentación.

La base de conocimiento del planificador contiene operadores que realizan las operaciones de presentación de: (1) generar texto usando plantillas de texto con información prefijada y variables que pueden ser instanciadas en tiempo de ejecución y (2) operadores para generar elementos gráficos (primitivas gráficas). Por ejemplo, las primitivas gráficas permiten ubicar un lugar en un mapa o anclar a un punto del mapa un texto o un gráfico de evolución que detalle información de interés de dicho punto en un instante dado.

El modelo de presentación también contiene las estrategias de cómo combinar y presentar segmentos de información relacionadas, que integran el resumen de comportamiento (patrones de discurso). En detalle, los patrones de discurso son estructuras de discurso parciales que expresan cómo presentar información, en función de condiciones sobre el estado del sistema hidrológico y/o necesidades de información. Por ejemplo, la parte superior de la Figura 1 ilustra gráficamente un patrón de discurso como un conjunto de enlaces entre objetivos comunicativos, donde cada objetivo comunicativo aporta piezas de información. Por ejemplo, la finalidad del objetivo comunicativo *elaborate\_with\_quantity(x, q)* es mostrar el valor de una cantidad  $q$ , para facilitar la interpretación del estado  $x$  de un componente simple a que está asociada.



**GOAL:** *elaborate\_with\_quantity(x, q)*  
**CONDITIONS:**  $(x: \textit{sección-de-río}) \wedge (q: \textit{caudal}) \wedge (\textit{value}(x, \textit{caudal}, \textit{current}, v) \wedge (t=0))$   
**MEDIA:** *text-body*  
**EFFECT:** *add\_text({"se registra un valor de ", v, " m<sup>3</sup>/seg en", c})*

---

**GOAL:** *contrast\_with\_previous\_value(x,q<sub>1</sub>)*  
**CONDITIONS:**  $(x: \textit{sección-de-río}) \wedge (q_1: \textit{caudal}) \wedge (\textit{value}(x, \textit{caudal}, \textit{previous}, w) \wedge (t=-1))$   
**MEDIA:** *text\_body*  
**EFFECT:** *add\_text({"la variación de caudal en ese punto del río es de ", |v-w|, " m<sup>3</sup>/seg con respecto a la hora anterior"})*

---

Figura. 1 Ejemplo de estructura de discurso, patrones de discurso y operadores [10].

Para informar sobre el estado de un componente se enlazan objetivos comunicativos para componer la estructura del discurso, ver

GOAL:	<i>elaborate_with_quantity(x, q)</i>
CONDITIONS:	$(x: \text{sección-de-río}) \wedge (q: \text{caudal}) \wedge (\text{value}(x, \text{caudal}, \text{current}, v) \wedge (t=0))$
MEDIA:	<i>text-body</i>
EFFECT:	<i>add_text({"se registra un valor de ", v, " m<sup>3</sup>/seg en", c})</i>
<hr/>	
GOAL:	<i>contrast_with_previous_value(x,q1)</i>
CONDITIONS:	$(x: \text{sección-de-río}) \wedge (q1: \text{caudal}) \wedge (\text{value}(x, \text{caudal}, \text{previous}, w) \wedge (t=-1))$
MEDIA:	<i>text_body</i>
EFFECT:	<i>add_text({"la variación de caudal en ese punto del rio es de ",  v-w , " m<sup>3</sup>/seg con respecto a la hora anterior"})</i>

Figura. 1. En el ejemplo que se muestra se enlazan *elaborate\_state*, *anticipate\_causes*, *elaborate\_causes*, *anticipate\_effects*, etc., Los objetivos comunicativos se realizan con operadores, como por ejemplo: para elaborar texto y alcanzar el objetivo comunicativo *elaborate\_state* se usan los operadores *elaborate\_with\_quantity(x, q)*, *contrast\_with\_previous\_value(x, q2)*, etc. [10]. Estos operadores realizan la descripción en texto usando recurrentemente la acción *add\_text* que enlaza segmentos de texto según condiciones de selección en cada operador para el instante actual ( $t=0$ ).

### 3.3.2. Modelo de Presentación Robot-Guía

En [77] se propone un modelo de presentación utilizando un Robot. Este Robot a nivel de software se distribuye como una aplicación distribuida, la versión más reciente es un agente basado en la arquitectura utilizando un enfoque específico COBRA como herramienta de integración. En la Figura 2 se muestra el proceso general y su arquitectura.

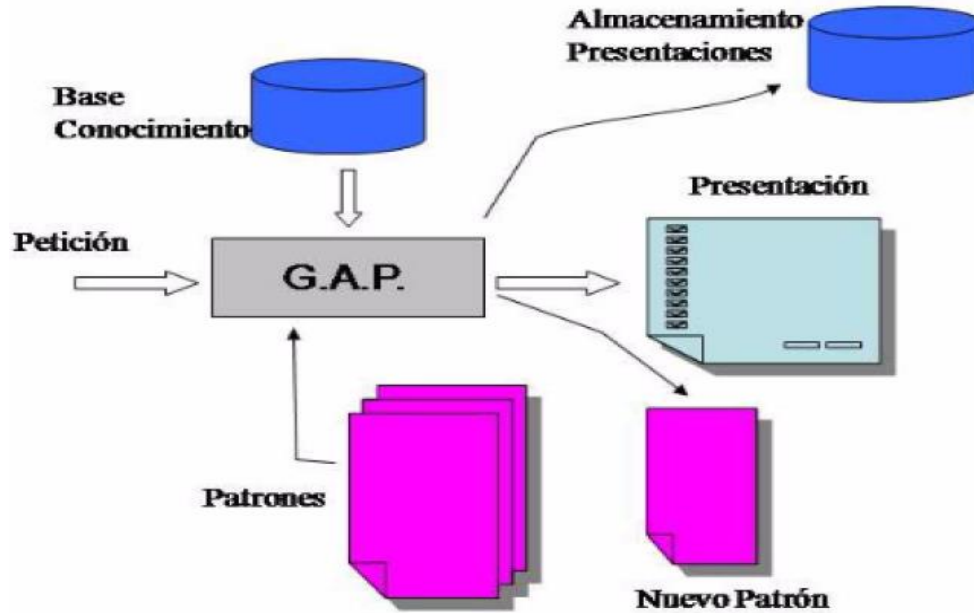


Figura 2 Arquitectura de la propuesta de Robot-Guía [77]

Teniendo en cuenta, por ejemplo, una solicitud, el sistema genera una presentación por parte del conjunto de patrones que mantienen. Es posible que existan varias presentaciones para el mismo patrón, porque una nueva presentación es guardada en un índice de éxito. El resultado es una estructura en árbol con diferentes alternativas.

Los procesos para obtener los contenidos de la presentación se realizan de forma repetitiva para cada tema (ítem o subítem) incluido en el patrón y cuenta con las siguientes fases:

- Fase de Selección: Es solicitado al servidor los posibles contenidos candidatos.
- Obtención del valor de inclusión: Para cada contenido candidato de cada tema se aplica el módulo de inferencia borrosa y se obtiene un valor de inclusión que se ha definido en tanto por ciento.
- Selección del mejor: Se toma el contenido de mayor inclusión y se coloca en la presentación.

### 3.3.3. Modelo de Presentación Encripnauta

El Encripnauta [85] es una aplicación desarrollada totalmente con Software Libre orientada a la creación de redacciones multimedia en línea. El objetivo de la herramienta es permitir la creación remota y cooperativa de presentaciones multimedia, integrando texto, imágenes y sonidos mediante el uso de sólo un navegador web estándar y una conexión a Internet.

El sistema se basa en una arquitectura cliente-servidor centralizada. La interfaz de usuario reside completamente en el navegador web. En el servidor residen todos los demás módulos: base de datos y middleware de acceso a datos, lógica de edición, gestor de contenidos (galería) y el generador de las presentaciones.

Una presentación o redacción multimedia está formada por diapositivas que a su vez contienen elementos multimedia configurables. Los objetos básicos almacenados en la base de datos son:

**Slide:** define cada “diapositiva” de una presentación. Una diapositiva contiene elementos de texto o imágenes descritos en las tablas Text e Img. Cada una de las diapositivas que compone una presentación siempre están relacionadas con una presentación.

**Text:** define cada uno de los elementos de texto de cada diapositiva. Los textos están definidos por la caja que los contiene, la posición inicial y final en la diapositiva, color de fondo, color de texto, tipo de letra, transición de entrada, transición de salida, etc. También pueden asignarse enlaces hipertexto (URL) a cada caja de texto.

**Img:** al igual que el texto, los gráficos están definidos por cajas, pero contienen imágenes, en cualquier formato gráfico, en vez de texto. En esta tabla se almacena la información de las imágenes que pertenecen a cada diapositiva, está relacionada con Slide y con Image. Sus atributos determinan el momento de inicio en la diapositiva, duración, posición, transparencia, así como sus transiciones de entrada y salida. Las imágenes deben estar previamente almacenadas en la

galería de imágenes. Se pueden incluir en una diapositiva siempre que se dispongan de los privilegios necesarios.

Transition: esta tabla define los efectos de transiciones que pueden aplicarse tanto a cajas de texto e imágenes como a las transiciones entre diapositivas. A efectos de que se puedan generar salidas con distintos formatos (SMIL, SWF, DOM/Javascript), las equivalencias son almacenadas en esta tabla.

Image: esta tabla almacena el contenido de las imágenes cargadas en la galería. La información almacenada incluye, propietario, tamaño, y los permisos de cada una de ellas.

Category: los elementos multimedia de la aplicación se dividen en categorías para facilitar la navegación y selección.

Presentation: mantiene la secuencia de diapositivas que conforman una presentación completa. Los elementos de esta tabla son creados por los usuarios con los privilegios adecuados. A cada presentación se le puede asignar un audio como música de fondo. Los ficheros de audios deben estar almacenados previamente en galería de audios.

User: esta tabla mantiene la información y privilegios de los usuarios registrados en el sistema. Cada presentación u elemento de la galería pertenece a un usuario.

Entity: en esta tabla se describen distintos centros docentes que participan en el concurso, cada usuario del sistema pertenece a una de las entidades.

# **CAPÍTULO IV: GENERACIÓN DE EXPLICACIONES EN SISTEMAS DINÁMICOS**

---

## **4.1. Visión General**

El objetivo en este trabajo está dirigido a plantear una solución técnica general para automatizar la generación y presentación de explicaciones de los resultados de la inferencia de un Sistema Experto de manera adecuada, utilizando diferentes medios de presentación de información. La explicación obtenida se basa en la descripción de los estados finales e intermedios por los que transita un Sistema Experto para llegar a una conclusión. El método que se propone debe tener la capacidad de interpretar la información según se presenta en el árbol de inferencia y consolidar una explicación coherente y entendible por los usuarios finales del sistema. Para garantizar el buen funcionamiento del método propuesto se debe contar con las siguientes entradas:

- Ontología desarrollada en Protégè. Esta ontología debe describir lo más ampliamente posible el dominio donde actúa el Sistema Experto, debe contener los conceptos y las relaciones identificadas en dicho dominio, así como las restricciones que pudieran existir. Los conceptos identificados y representados en la ontología deben coincidir con los objetos identificados en el modelo de objetos utilizados por el Sistema Experto para el desarrollo de las reglas.
- Árbol de inferencia. Este árbol debe ser un fichero que contenga el estado final y los estados intermedios por los que transitó el Sistema Experto para llegar a la conclusión final. Cada uno de los estados intermedios debe contar con los valores que gatillaron el resultado obtenido.
- Patrones de presentación. Estos patrones deben ser adecuados para el dominio, es decir, ajustarse a las necesidades de información de los destinatarios, para crear a partir de ellos las presentaciones.

También el sistema necesita cumplir con las siguientes restricciones:

- El Sistema Experto debe utilizar Drools [86] como gestor de las normas, operaciones, definiciones y restricciones brindadas por los expertos del dominio.
- El Sistema Experto debe caracterizar a un sistema dinámico complejo. Este sistema dinámico debe ser gestionado por el hombre y se requiere de él información adecuada para apoyar a la toma de decisiones.

La Figura 3 muestra la arquitectura general del sistema. En detalle se tiene un Sistema Experto y dentro de éste se cuenta con una base de reglas que caracteriza su funcionamiento. El Sistema Experto en tiempo de ejecución genera un árbol de inferencia construido por Drools. De este árbol de inferencia interesa para la presente propuesta los estados intermedios representados en cada nodo del árbol y el estado final donde ambos forman parte de las entradas para el método que implementa la propuesta. El resto de los componentes que se pueden ver en la Figura 3 serán descritos a continuación.

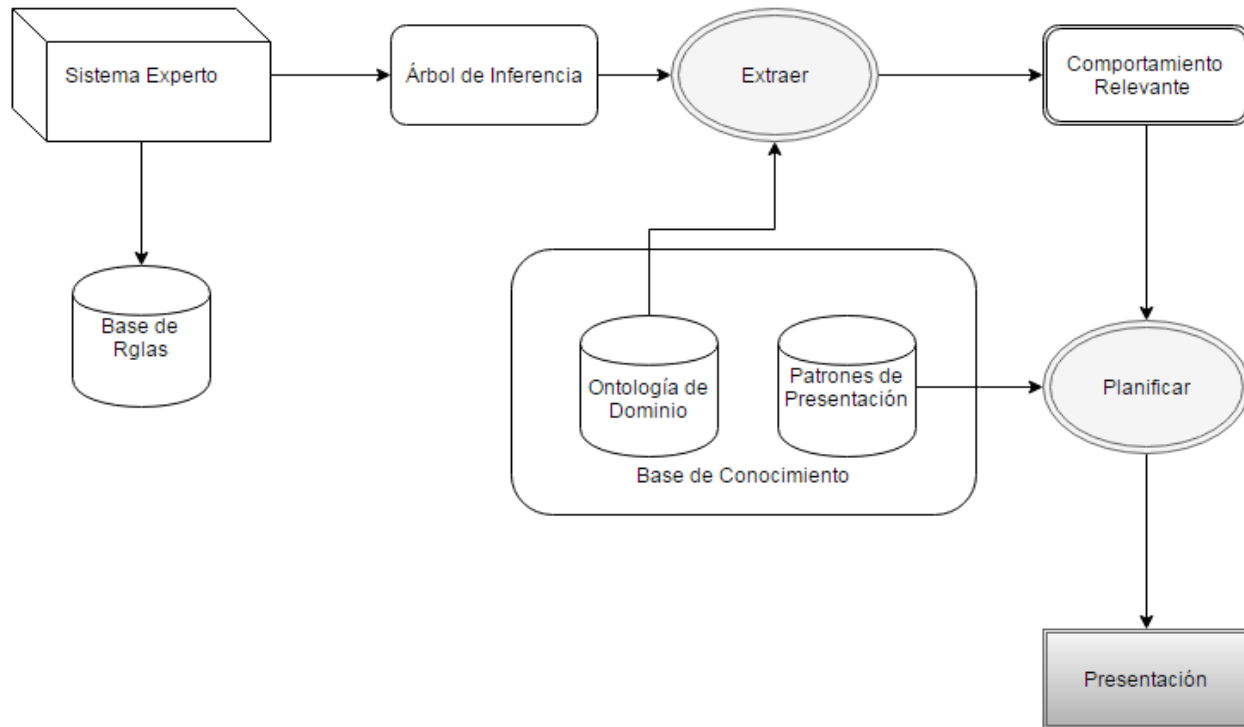


Figura 3. Arquitectura General de la Propuesta (las elipses representan tareas, los cilindros representan conocimiento previo, los rectángulos representan datos).

De forma específica, la Figura 3 muestra los elementos fundamentales que componen la solución propuesta. Se trata de un método compuesto por dos tareas principales que hacen uso de la base de conocimiento del dominio. En detalle estas tareas se describen como:

**Extraer:** Tarea que recibe como entrada el conjunto de reglas que fueron activadas por el Sistema Experto (árbol de inferencia) para llegar a la conclusión y la ontología del dominio. Esta tarea brinda como salida, después de un proceso de enlace entre la ontología y las reglas disparadas por el Drools, información que será definida como comportamiento relevante.

**Planificar:** Tarea que recibe el comportamiento relevante que se obtuvo como salida de la tarea Extraer y que será preparada para presentar. Esta tarea cuenta con el conocimiento de los patrones de presentación y los elementos discursivos, que junto con la entrada (comportamiento relevante generado con la tarea Extraer) se usarán para generar un plan de presentación. Finalmente, de la tarea Planificar se obtiene la explicación del Sistema Experto en un formato de presentación multimedia.

Para facilitar el entendimiento de la propuesta en las siguientes secciones se detallan, la visión general del dominio de trabajo (en el apartado "Visión General del Sistema Dinámico) y las tareas **Extraer** y **Planificar**. En cada una de las secciones que explican las tareas (Extraer y Planificar) serán presentados los pseudocódigos que describen la dinámica del método propuesto.

## 4.2. Visión General del Sistema Dinámico

El presente trabajo contempla un tipo de sistema dinámico que presenta ciertas características:

- Opera en un contexto de gestión de acuerdo con objetivos prefijados. Estos objetivos pueden estar orientados a mantener el sistema en ciertos niveles de explotación, pueden estar orientados a mantener el sistema bajo ciertos umbrales que no afecten a las personas, etc.
- Es un sistema físico que es gestionado por personal humano donde es importante contar con información adecuada y a tiempo para ayudar a la toma de decisiones.

Estas decisiones pueden a su vez incidir en el comportamiento futuro del sistema, que, además, puede tener impacto en personas o empresas, así como en la ciudad o país donde esté situado el sistema. Un ejemplo de este tipo de sistemas es una minera, donde se forman pilas preparadas para ser explotadas mediante diferentes técnicas como por ejemplo la Biolixiviación, donde existen operadores,

investigadores expertos y directivos de la empresa, que obran en consecuencia; interesados por tanto en la evolución del sistema.

Otros ejemplos de sistema de este estilo son: tráfico en una red de carretera, en este sistema el interés de los que dirigen el tráfico está orientado a mantener las carreteras sin atascos, pudiendo tomar la decisión de reorientar la circulación de vehículos antes fenómenos no esperados o en horarios de mayor circulación. La red de plantas generadoras de energía eléctrica es otro sistema de este estilo donde los operarios según lo observado pueden tomar decisiones.

Los sistemas dinámicos a los que se adecúa la propuesta deben contar con las siguientes características:

- Sistemas estructuralmente complejos con cantidades importantes de elementos relacionados entre sí.
- Sistemas monitorizados mediante centros de control para adecuar (en caso de que sea necesario) su comportamiento a objetivos de gestión predeterminados.
- Sistemas con comportamiento medido a través de redes de sensores, cuyas condiciones son analizadas por expertos humanos para establecer funcionamiento (actual o probablemente en el futuro).

Para lograr representar adecuadamente el sistema se consideran los siguientes objetivos:

- Lograr capturar en la representación la complejidad del sistema a un nivel adecuado de detalle para alcanzar los objetivos.
- El conocimiento a representar debe ser asequible de adquirir y formalizar para evitar esfuerzos demasiado elevados que impliquen costes no asumibles.

Se plantea una forma de representación ontológica como una alternativa adecuada por razones como las siguientes:

- Definen los términos básicos y en función de ellos, se esbozan las relaciones que se pueden establecer en ese dominio y las reglas para su combinación.
- Contemplan las restricciones (slots) para delimitar las cualidades de los conceptos y las facetas como expresión de los valores que puede adoptar un slot.
- Establecen las instancias como objetos de la clase.

#### **4.2.1. Representación del sistema dinámico**

Para formalizar la representación del sistema dinámico en el presente trabajo se utiliza una ontología. En concreto, se maneja una ontología de dominio ya que permite representar el conocimiento especializado de un determinado campo, como lo son los procesos mineros.

El uso de la ontología proporciona una forma de representar y compartir el conocimiento utilizando un vocabulario común. Además, usa un formato de intercambio de conocimiento que facilita la interacción con otros sistemas proporcionando un protocolo específico de comunicación y reutilizando el conocimiento.

Para describir el conocimiento representado en la ontología el método propuesto utiliza RDF/XML. El RDF es un lenguaje general utilizado para descomponer conocimiento en piezas pequeñas (triples), teniendo en cuenta reglas acerca de la semántica o significado de esas piezas [87]. La base del procesamiento de este archivo en formato RDF es descomponer el conocimiento en un grafo dirigido etiquetado. Cada arista en el grafo representa un hecho, o relación entre dos cosas. Un hecho representado de esta manera tiene tres partes: un sujeto, un predicado y un objeto. El sujeto es lo que está al inicio de la arista, el predicado es el tipo de arista y el objeto es lo que está al final de la arista.

Notation 3 (N3), es el estándar de facto para escribir RDF. Las sentencias son escritas como la URI del sujeto (entre paréntesis angulares "<>"), seguida de la

URI del predicado, seguida por la URI del objeto o un valor literal, seguida de un punto. El RDF es codificado mediante un formato XML, dado que trabajan bajo el mismo modelo abstracto de N3.

En un documento RDF/XML hay dos tipos de nodos XML que se describen como:

- **Nodos de recursos XML.** Los nodos de recursos XML son los sujetos y los objetos de las sentencias, y ellos usualmente son etiquetas del tipo `rdf:Description` que tienen un atributo `rdf:about` indicando la URI del recurso al cual representan. Los nodos de recursos XML contienen en sí nodos de propiedad XML.
- **Nodos de propiedades XML.** Cada nodo de propiedad representa una sola relación. El sujeto de esa relación es el nodo externo que contiene dicha propiedad. Las relaciones pueden tener como objetos recursos o valores literales. Para poner un valor literal como el objeto de un nodo de propiedad XML, el valor va simplemente dentro del elemento. El predicado `rdf:type` es usado para decir de qué tipo es el recurso.

El fichero RDF/XML generado tras el modelado de la ontología, puede ser fácilmente dividido en secciones que corresponden a cada uno de los componentes de la ontología: conceptos o clases, relaciones o propiedades e instancias o individuos.

Los individuos representan objetos del dominio de interés y son representados en el archivo XML como se muestra en la Figura 4a. Las propiedades son relaciones binarias sobre los individuos y pueden ser inversas, transitivas o simétricas. Existen dos tipos de propiedades:

- *ObjectProperties*, permite relacionar un individuo con otro (véase Figura 4b), la etiqueta que identifica un `objectProperty` es `<owl:objectproperty>`. La propiedad *ObjectProperties* debe tener 3 elementos: el primero es el nombre, el segundo es el dominio que hace referencia a la clase o clases

iniciales y el tercero es el rango que hace referencia a la clase o clases finales.

- *DataProperties*, permite relacionar a un individuo con su valor (véase Figura 4c), la etiqueta que identifica un objectProperty es <owl:datatypeproperty>.

Por último, las clases se entienden como conjuntos que contienen individuos y pueden ser organizadas dentro de una jerarquía de clases y subclases, esto se conoce como una taxonomía. Las clases también son conocidas como conceptos, pues son una representación concreta de éstos. La Figura 4d muestra como son representadas las clases en el archivo XML.

```

<!--
////////////////////////////////////
// Individuals
////////////////////////////////////

<!-- http://www.semanticweb.org/...#A1 -->

<owl:NamedIndividual rdf:about="http://www.semanticweb.org/...A1">
  <rdf:type
rdf:resource="http://www.semanticweb.org/...#Termofilos_Moderados"/>
</owl:NamedIndividual>
    
```

Figura 4a. Ejemplo de representación de individuos en el XML.

```

<!--
////////////////////////////////////
// Object Properties
////////////////////////////////////

<!-- http://www.semanticweb.org/...#Property -->

<owl:ObjectProperty rdf:about="http://www.semanticweb.org/...#Concept1">
  <rdfs:domain rdf:resource="http://www.semanticweb.org/...#Concept2"/>
  <rdfs:range rdf:resource="http://www.semanticweb.org/...#Concept3"/>
  <rdfs:range rdf:resource="http://www.semanticweb.org/...#Concept4"/>
  ...
</owl:ObjectProperty>
    
```

Figura 4b. Ejemplo de representación de la propiedad ObjectProperties en el XML.

```

<!--
////////////////////////////////////
//Data properties
////////////////////////////////////

<!--http://www.semanticweb.org/yahima...#DataProperty1 -->

<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/...# DataProperty1">
  <rdfs:domain rdf:resource="http://www.semanticweb.org/...#Concept1"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/...# Concept2"/>
    
```

```

<rdfs:domain rdf:resource="http://www.semanticweb.org/...# Concept3"/>
...
</owl:DatatypeProperty>

```

Figura 4c. Ejemplo de representación de la propiedad DataProperties en el XML.

```

<!--
////////////////////////////////////
// Classes
////////////////////////////////////

  <!-- http://www.semanticweb.org/...#Concept1 -->
<owl:Class rdf:about="http://www.semanticweb.org/...# Concept1">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/...# Concept2"/>
  <owl:disjointWith rdf:resource="http://www.semanticweb.org/...# Concept3"/>
</owl:Class>

  <!-- http://www.semanticweb.org/...#Concept4 -->
<owl:Class rdf:about="http://www.semanticweb.org/...# Concept4">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/...#Concept5"/>
  <rdfs:comment>Comment1</rdfs:comment>
  ...
</owl:Class>
...

```

Figura 4d. Ejemplo de representación de una clase en el XML.

Figura 4. Representación del XML de una ontología desarrollada en Protègè.

Las figuras 4a, 4b, 4c y 4d corresponden a descripciones de segmentos de un mismo archivo RDF/XML que representa la ontología de dominio. Cada elemento caracterizado en la ontología va a contener en su slot descripciones de texto en lenguaje natural que se va a utilizar para describir el elemento atómicamente y luego unirlo con otras descripciones conformando de esta manera la explicación. Dichas descripciones corresponden a las plantillas de texto que serán comentadas a continuación.

#### 4.2.2. Representación del Sistema Experto

El Sistema Experto está caracterizado por las reglas obtenidas de los expertos en el dominio en el cual se quiere desarrollar el sistema. Estas reglas son materializadas con Drools y para obtener lo que sucede internamente en el motor de regla lo primero que se debe hacer es modificar el código del Sistema Experto.

Drools admite la adición de dos tipos de registradores de *KieSessions*: *ConsoleLogger* y *FileLogger* [88].

El *KieRuntimeLogger* utiliza el sistema integral de eventos en Drools para crear un registro de auditoría que se puede utilizar para registrar la ejecución de una aplicación para una inspección posterior, utilizando herramientas tales como el visor de auditoría de Eclipse. Este módulo permite configurar estos registradores de *KieSessions* utilizando etiquetas XML. Estas etiquetas tienen los nombres: `<kie: consoleLogger ...>` y `<kie: FileLogger...>` [88], [89]. Esta propuesta utiliza la forma `<kie: FileLogger...>` para obtener el registro de auditoría del Sistema Experto que no es más que el árbol de inferencia. La Figura 5 muestra las líneas de código que posibilitan la obtención del fichero Logger.

```
KieRuntimeLogger logger =
    KieServices.Factory.get().getLoggers().newFileLogger(ksession,
        "logdir/mylogfile");
...
logger.close();
```

Figura 5. Líneas de código para obtener el FileLogger.

Al ejecutar el Sistema Experto con esta nueva adición, se obtiene un fichero que contiene un evento de activación registrado por el *WorkingMemoryLogger*. La Figura 6 muestra un fragmento de la estructura del fichero obtenido.

```
<object-stream>
<org.drools.core.audit.WorkingMemoryLog>
  <version>6.1</version>
  <events>
    <org.drools.core.audit.event.ObjectLogEvent>
      <type>1</type>
      <factId>1</factId>
      <objectToString>Hecho1[value=000]</objectToString>
    </org.drools.core.audit.event.ObjectLogEvent>
    ...
    <org.drools.core.audit.event.ActivationLogEvent>
      <type>4</type>
      <activationId>Nombre de la regla[listaIdentificadores]</activationId>
      <factHandleIds>Identificadores de hechos activados</factHandleIds>
      <rule> Nombre de la regla </rule>
    </org.drools.core.audit.event.ActivationLogEvent>

    <org.drools.core.audit.event.ActivationLogEvent>
      <type>6</type>
      <activationId>Nombre de la regla[listaIdentificadores]</activationId>
      <rule> Nombre de la regla </rule>
      <declarations>Conclusión=tempConclusión[value=0000]</declarations>
```

```

    <factHandleIds>Identificadores de hechos activados </factHandleIds>
  </org.drools.core.audit.event.ActivationLogEvent>
  ...
  <org.drools.core.audit.event.ObjectLogEvent>
    <type>2</type>
    <factId>1</factId>
    <objectToString>TempEstimada[value=0000]</objectToString>
  </org.drools.core.audit.event.ObjectLogEvent>
  ...
  <org.drools.core.audit.event.ActivationLogEvent>
    <type>7</type>
    <activationId> Nombre de la regla[listaIdentificadores]</activationId>
    <rule> Nombre de la regla</rule>
  </org.drools.core.audit.event.ActivationLogEvent>
  ...
  <org.drools.core.audit.event.ObjectLogEvent>
    <type>3</type>
    <factId>1</factId>
    <<objectToString>TempEstimada[value= inicial]</objectToString>
  </org.drools.core.audit.event.ObjectLogEvent>
  ...
  </events>
  <engine>PHREAK</engine>
</org.drools.core.audit.WorkingMemoryLog>
</object-stream>

```

Figura 6. Fragmento del fichero WorkingMemoryLogger obtenido.

Un evento activado, registrado por el WorkingMemoryLogger, es una instantánea del proceso llevado a cabo por Drools, anteriormente contenido en la memoria de trabajo. Dicho evento contiene el identificador de activación, el nombre de la regla y una cadena que representa las declaraciones de la activación. Estas declaraciones son una lista de pares nombre-valor para cada una de las declaraciones de las tuplas de la activación. El nombre es el identificador de la declaración, y el valor son un método *toString* del valor del parámetro, seguido de la identificación del hecho entre corchetes ([...]), por ejemplo, param1 = 10; param2 = Persona [John Doe].

#### 4.2.2.1. Descripción del evento WorkingMemoryLogger

El fichero WorkingMemoryLogger cuenta con diferentes etiquetas que la caracterizan. Dentro de estas etiquetas se puede obtener el detalle de cada registro de auditoría. A continuación, son descritas cada una de las etiquetas de interés para la propuesta.

Bajo las etiquetas `<object-stream>` y `<org.drools.core.audit.WorkingMemoryLog>` es descrito todo el registro de auditoría. Los primeros elemento del registro corresponden a la inserción de los hechos y estos son identificados por estar bajo la etiqueta `<org.drools.core.audit.event.ObjectLogEvent>`. Cada uno de los hechos insertados son reconocidos por un identificador único (`<factId>1</factId>`).

Al tener todos los hechos necesarios para activar una regla aparece la etiqueta `<org.drools.core.audit.event.ActivationLogEvent>`, bajo esta etiqueta se describen todos los detalles de la regla disparada. Es importante comentar que en la etiqueta `<activationId>` seguido del identificador de la regla dentro de corchetes ([ ]) se encuentra una lista con los identificadores de los hechos utilizados para disparar la regla. Dentro de la etiqueta `<declarations>` son creados los objetos necesarios para disparar la regla.

Al terminar las ejecuciones de activación Drools procede a la actualización de objeto en el registro. Es importante destacar que las ejecuciones no están en el orden en que se crearon, sino en el orden que Drools juzgue apropiado cuando resuelva los conflictos.

Por último, son eliminados los objetos de la memoria de trabajo. La eliminación está bajo la etiqueta `<org.drools.core.audit.event.ObjectLogEvent>` seguida por `<type>3</type>`.

Después de conocer la estructura interna del `WorkingMemoryLogger` y la manera de obtener el árbol de inferencia de Drools, así como la estructura del RDF donde está almacenada la ontología que describe el dominio, se puede proceder a la explicación de las dos tareas que se proponen para este modelo.

### 4.3. Tarea Extraer

La tarea **Extraer** es la primera tarea que se ejecuta dentro del método propuesto. Esta tarea es la encargada de vincular los archivos RDF/XML obtenido de la ontología y el fichero WorkingMemoryLogger obtenido como resultado de la inferencia que se obtiene con Drools. Estos archivos corresponden a la descripción del dominio específico de aplicación. Como resultado la tarea Extraer devuelve un conocimiento que se ha denominado comportamiento relevante.

Este comportamiento relevante está compuesto por cada uno de los elementos activados por la ejecución de una regla y un comentario. El comentario contiene la descripción puntual de cada uno de los elementos, así como sus propiedades y restricciones. Esta información está lista para ser recibida por el planificador y obtener una explicación de lo ocurrido.

De manera simplificada puede decirse que el comportamiento relevante son los elementos del RDF que fueron identificados como activados en el fichero WorkingMemoryLogger. Este proceso se describe como un enlace de los elementos del WorkingMemoryLogger con sus homólogos en el fichero que describe la ontología del dominio. Para facilitar el entendimiento de la tarea Extraer en la Figura 7 se presenta el pseudocódigo que describe su actuar.

<p><b>Program:</b> Extraer</p> <p><b>Environment:</b> ÁRBOL_INFERENCIA (archivo XML) y ONTOLOGÍA (archivo RDF/XML)</p> <p><b>Algorithm:</b></p> <ol style="list-style-type: none"> <li>1. Extracción de hechos (ÁRBOL_INFERENCIA)</li> <li>2. Identificación de hechos y reglas activadas (ÁRBOL_INFERENCIA)</li> <li>3. Extracción de elementos activados(ListFacts)</li> <li>4. Buscar elementos activados en RDF/XML(Objects, Characteristics, ONTOLOGÍA)</li> <li>5. Planificador(Objects ,Characteristics y Matriz_Valor)</li> </ol> <p><b>End Algorithm</b></p>
---

Figura 7. Pseudocódigo de la tarea Extraer

(1) **Extracción de hechos:** Este método es el encargado de recorrer el WorkingMemoryLogger en busca de cada uno de los hechos ahí presentes. Se conoce que la declaración de los hechos es realizada entre las etiquetas <objectToString>. Cada uno de estos hechos es manipulado utilizando un identificador el cual corresponde al declarado en el fichero WorkingMemoryLogger bajo la etiqueta <factId>. Como resultado de la ejecución del método se obtiene una lista con todos los hechos que se reflejan en el WorkingMemoryLogger. La Figura 8 muestra el pseudocódigo del método.

```

Program: Extracción de hechos
Environment: ÁRBOL_INFERENCIA (archivo XML)
Algorithm:
  While XML != "":
    Read línea
    If "<factId>" in línea
      Hecho.id = ID
    If "<objectToString>" in línea
      Save Hecho en lista List
      Add List a ListFacts
    End If
  End While
  Return (ListFacts)
End Algorithm

```

Figura 8. Pseudocódigo del método "Extracción de los hechos del árbol de inferencia".

(2) **Identificación de hechos y reglas activadas:** Este método recorre el WorkingMemoryLogger en busca de cada regla activada. Se conoce que la activación de una regla es realizada entre las etiquetas <activationId> y antes de cerrar la etiqueta dentro de corchetes ([]) se encuentran los identificadores (ID) de los hechos que dieron lugar a la activación de la regla. Es importante el orden en que aparecen los ID de estos hechos ya que es el orden en que fueron llamados por la regla. Con los identificadores de los hechos se recuperan los elementos activados de cada uno de dichos hechos y se procede a activar la explicación. Se espera obtener después de la ejecución del método los hechos activados y el orden en que fueron activados. La Figura 9 muestra el pseudocódigo del método.

```

Program: Identificación de hechos y reglas activadas
Environment: ÁRBOL_INFERENCIA (archivo XML)
Algorithm:
  While XML != "":
    Read línea
    If "<activationId> " in línea
      Get ID de hechos
      Save ID in ActivationOrder
    End If
  End While
  Return (ActivationOrder)
End Algorithm

```

Figura 9. Pseudocódigo del método "Identificación de hechos y reglas activadas".

(3) **Extracción de elementos activados:** Este método es el encargado de separar de ListFacts cada objeto y sus características teniendo en cuenta que los ID de los hechos se encuentren en ActivationOrder. Los objetos son capturados ya que se encuentran inmediatamente después de la etiqueta <objectToString> y terminan antes del próximo corchete (]). Además, son considerados elementos activados cada una de las características del hecho. Estas características son identificadas ya que se encuentran secuencialmente dentro de corchetes ([]) y separadas entre sí por una coma (,). La Figura 10 muestra el pseudocódigo del método.

```

Program: Extracción de elementos activados
Environment: LIST_FACTS (lista), ACTIVATION_ORDER(lista)
Algorithm:
  Matriz_valor = new matriz inicializada en 0
  Objects = []
  Characteristics = []
  For i range (len (ActivationOrder)):
    For x in ListFacts:
      If(ListFacts[x].id == ActivationOrder[i]
        Object = Get objeto de ListFacts[x]
        Objects = Add (Object)
        Characteristic = Get características de ListFacts[x]
        Characteristics = Add (Object)
        col = Objects.index(Object)
        fila = Characteristics.index(Characteristic)
        Matriz_valor [col][fila] = Dato de la
          Característica
      End IF
    End For
  End For

```

```

End For
Return (Objects, Characteristics, Matriz_valor)
End Algorithm

```

Figura 10 Pseudocódigo del método "Extracción de elementos activados".

(4) Buscar elementos representativos en RDF/XML: Este método es el encargado de empatar cada uno de los elementos activados con el RDF/XML. Primero enlazamos cada uno de los elementos de la lista *Objects* con cada uno de sus homólogos en el RDF/XML. Estos homólogos son buscados dentro de la sección donde son declaradas las clases del dominio. Por cada uno de los elementos encontrados tomaremos la sección de comentario la cual encontramos bajo la etiqueta `<rdfs:comment>`. Cada elemento cuenta en su sección *comment* con una o más descripciones en lenguaje natural que serán utilizadas para elaborar la explicación. Finalmente, el método devuelve una lista con los comentarios de cada uno de los elementos que contiene la lista *Objects*. La Figura 11 muestra el pseudocódigo del método.

```

Program: Buscar elementos representativos en RDF/XML
Environment: Objects(lista), Characteristics(lista) y
ONTOLOGÍA (RDF/XML)
Algorithm:
While XML != "":
  Read línea
  For i in Objects:
    If línea == <owl:Class rdf:about=
      "http://www.semanticweb.org/...# Objects[i]>
      Comment = Get argumento de <rdfs:comment>
      Comments = Add(Comment)
    End If
  End For
  Return (Comments)
End While
End Algorithm

```

Figura 11. Pseudocódigo del método "Buscar elementos representativos en RDF/XML".

#### 4.4. Tarea Planificar

La tarea **Planificar** es la encargada de reunir los criterios que permiten construir automáticamente una presentación que muestra la información correspondiente a las reglas que se activaron al ejecutar el Sistema Experto. Para presentar dicha información que explicaría el "por qué" un Sistema Experto arrojó un determinado resultado, es útil combinar formas de presentación como las siguientes:

- Texto en lenguaje natural, para expresar de una manera uniforme los contenidos principales del árbol de inferencia arrojado por Drools. Esta forma de expresión resulta muy conveniente para propósitos generales de comunicación y facilita el entendimiento cuando es proporcionada una explicación.
- Elementos gráficos para apoyar la explicación, facilitando el entendimiento del comportamiento de los elementos del sistema. Por ejemplo, gráficos con evoluciones temporales para expresar de forma detallada los valores numéricos de variables, vídeos, fotografías, etc.

La tarea en cuestión promete un esquema de presentación de la información con el objetivo de establecer una explicación apta sobre las reglas activadas en el Sistema Experto. Es decir, se trata de usar formas de presentación habituales que permitan vincular las diferentes piezas de información, y que el conjunto resulte cercano y convincente para el usuario. A tal efecto se plantean los siguientes objetivos:

- *Natural*. Se trata de que la explicación utilice descripciones de funcionamiento del sistema dinámico en la forma natural en que lo entienden las personas con el fin de que la explicación sea comprensible.

- *Convincente*. En la medida de lo posible, la explicación presentada debe convencer al usuario sobre la relevancia de la información mostrada, suministrando información adicional en forma de evidencias.

Tomando como base el modelo propuesto por Flores [10] en su tesis doctoral, se ha planteado un método que construye una presentación de forma dinámica, combinando texto y elementos gráficos. No se tiene una forma de presentación prefijada general para todos los casos, tal como se suele suceder en los sistemas informáticos convencionales. Por el contrario, con la solución que se plantea, cada presentación puede ser diferente y adaptada a las necesidades de comunicación de cada caso.

Esta propuesta abordó la generación dinámica de presentaciones con un **planificador** que maneja una base de conocimiento del dominio con dos tipos de elementos principales:

- *Operadores de presentación*. Para llevar a cabo las diferentes formas de presentación se utilizan varios tipos de operadores de presentación. Por ejemplo, se manejan operadores de plantillas de texto que aportan patrones de frases. Además, se manejan otros operadores para presentaciones gráficas.
- *Patrones de discurso*. Se trata de estructuras parciales de discurso generales que expresan cómo presentar información en función de condiciones sobre el estado del sistema dinámico y el destinatario de la presentación. Los patrones de discurso establecen estrategias generales de formas de presentar información.

A continuación, se describen en detalles los operadores de presentación y los patrones de discurso.

#### 4.4.1. Operadores de presentación

Los operadores de presentación establecen las formas básicas de presentación. Un tipo de operador de presentación es el que proporciona plantillas de frases. Las plantillas de frase son unidades de texto (segmentos de texto con variables para sustituir con información específica) a partir de los cuales se generan los textos explicativos.

Estas plantillas son frases en lenguaje natural diseñadas para comunicar de forma natural la secuencia de reglas disparadas por el Sistema Experto para llegar a una conclusión. Las variables de las plantillas de texto se instancian en tiempo de ejecución con valores concretos. La Figura 12 muestra ejemplos de plantillas de texto en el dominio de Biolixiviación.

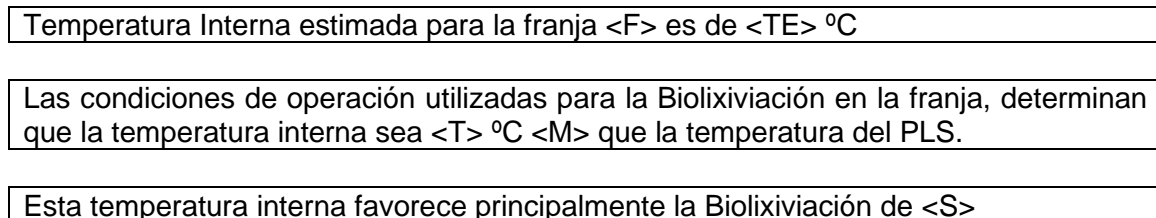


Figura 12 Ejemplos de plantillas de frases

En la Figura 12:

- *F* y *S* son datos que instancian valores cuando se consulta al Sistema Experto.
- *T* es un valor que se ha calculado utilizando las propiedades que caracterizan a los conceptos del dominio (en este caso: temperatura interna y temperatura del PLS).
- *M* es un valor inferido por el Sistema Experto.
- [TM] es una lista que será poblada con conceptos de la ontología del dominio (en este caso tipos de microorganismos).

En una correspondencia de estas plantillas con el Sistema Experto para la estimación de la temperatura interna en una pila específica se puede obtener un texto como el de la Figura 13:

**Las condiciones de operación utilizadas para la Biolixiviación en la franja 508 determinan una temperatura interna de 20 °C.**

La temperatura interna obtenida se debe a que la suma de los microorganismos es mayor que  $10e5$ , para este caso específico es de 1608084. Se puede decir que la temperatura interna estimada alcanzó el valor declarado ya que la temperatura del PLS superó los 20 °C , en este caso la temperatura del PLS fue de 25,4 °C.

Figura 13. Ejemplo de explicación generada con plantilla de frases.

Además, se ha considerado para esta propuesta operadores gráficos. Entiéndase por operadores gráficos los recursos didácticos que permiten comprender las causas y consecuencias de los eventos expuestos o narrados mediante documentos escritos por medio de recursos educativos digitales. Anders Bouwer [90] ofrece otra definición de operadores gráficos, plantea que pueden entenderse como unidades elementales que componen una visualización y que pueden entenderse atributos gráficos distinguibles como tamaño, color, posición, dirección, etc..

Para esta propuesta los operadores gráficos pueden ser instanciados en gráficos de tipo barra, línea, 2D, 3D etc., los cuales son utilizados para mostrar comparaciones entre valores. La Tabla 5 describe la directiva a utilizar. Estas directivas se generan por interpretaciones de preferencias de los usuarios respecto a cómo distribuir los datos en la gráfica. Ejemplos de estas interpretaciones pueden ser el tipo de gráfico en el que deben volcar los datos, orientación, dimensión, etc.

Tabla 5 Directivas para la gráfica

Nombre de la directiva	Descripción
makeBarGraph(x,z,y,w)	Esta directiva activa una función que construye un gráfico de barra donde cada uno de los parámetros tiene el significado siguiente:

	x: orientación de las barras (horizontales o verticales) z: rótulo del eje x y: rótulo del eje y w: dimensión (3D o 2D)
makePieGraph(x,w)	Esta directiva activa una función que construye un gráfico pastel donde cada uno de los parámetros tiene el significado siguiente: x: diseño del gráfico w: dimensión (3D o 2D)
makeLineGraph(x,z,y,w)	Esta directiva activa una función que construye un gráfico de línea donde cada uno de los parámetros tiene el significado siguiente: x: diseño del gráfico z: rótulo del eje x y: rótulo del eje y w: dimensión (3D o 2D)

#### 4.4.2. Patrones de Discurso

Un patrón de discurso expresa cómo construir una parte de la presentación. La estructura del discurso puede entenderse como la forma de disponer de principio a fin la información en nodos de presentación (como por ejemplo texto, imágenes, etc.) y de combinar estos nodos de presentación, de forma que faciliten la comprensión de la información [78]. Los patrones se formulan como una secuencia de objetivos comunicativos. Entiéndase por objetivo comunicativo el propósito que tiene el sistema de transmitir información al usuario. Para esta propuesta se utiliza la narración como estructura discursiva y dentro de ésta es utilizado el estilo periodístico.

La presentación completa está formada por la selección de varios patrones de discurso escogidos en tiempo de ejecución. Cada representación de un patrón de discurso es encabezada por un objetivo comunicativo. Un patrón puede contener:

- Objetivo, está relacionado con el objetivo comunicativo principal o general de la información.
- Condiciones, estas son planteadas mediante funciones.
- Media, es donde se declaran la forma de presentación a utilizar.
- Cuerpo de la presentación.

Un ejemplo de patrón es el siguiente:

<b>OBJETIVO:</b>	<i>informar_titular(x)</i>
<b>CONDICIONES:</b>	<i>existe(x) ^ final(x)</i>
<b>MEDIA:</b>	<i>plantilla-texto</i>
<b>CUERPO:</b>	<i>{ "Temperatura interna estimada es" &lt;f&gt;"°C, para la franja" &lt;c&gt; }</i>

Este patrón de discurso tiene como objetivo comunicativo global informar sobre la conclusión a la que arribó el Sistema Experto. En este patrón se manejan el operador de presentación *plantilla-texto* descrito anteriormente.

Cada patrón puede incluir un conjunto de condiciones para ser seleccionado, en este caso fueron llamadas las funciones *existe(x)* y *final(x)*. La función *existe(x)* verifica que existe un elemento *x* en la lista *Objects* que forma parte del comportamiento relevante obtenido por la tarea Extraer, mientras que la función *final(x)* verifica que el identificador del elemento *x* sea el último de la lista *ActivationOrder*. El valor de la variable *f* es obtenida de la matriz *Matriz\_valor*, mientras que la variable *c* es un dato que se captura en el proceso.

Cuando se trata de objetivos comunicativos que se descomponen en subobjetivos, se debe manejar el *discurso*. El siguiente ejemplo muestra un patrón que maneja *discurso* de forma recursiva. En este patrón son utilizadas las funciones *causa(y,x)* y *efecto(x)*. La primera función, *causa(y,x)*, verifica que en la lista *ActivationOrder* (lista que contiene el orden de activación) el identificador de *x* se encuentre inmediatamente después del identificador de *y*. La función *detallar\_caracteristica(x,y)*, se encarga de activar la descripción que enlaza los elementos *x* e *y* de acuerdo a sus características. Esta descripción está contenida dentro de la lista *Characteristics*.

**OBJETIVO:** detallar\_estado(x)  
**CONDICIONES:** causa(y,x), efecto(x)  
**DISCURSO:** detallar\_estado(y),  
 detallar\_caracteristica(x,y)

En la Tabla 6 se encuentran todos los patrones de discursos que se implementaron para la propuesta.

Tabla 6. Patrones de Discurso

<b>Patrón Titular 1</b>	
OBJETIVO:	informar_titular(x)
CONDICIONES:	existe(x) ^ final(x)
MEDIA:	plantilla-texto
CUERPO:	{"Temperatura interna estimada es" <f>"°C, para la franja" <c>}
<b>Patrón Titular 2</b>	
OBJETIVO:	sin_información ()
MEDIA:	plantilla-texto
CUERPO:	{"No existe información suficiente para obtener una conclusión válida"}
<b>Patrón Detallar Estado Simple</b>	
OBJETIVO:	detallar_estado(x)
CONDICIONES:	causa(y,x), efecto(x), NOT causa(x,_)
MEDIA:	plantilla-texto
CUERPO	{" La temperatura interna alcanzada favorece principalmente la Biolixiviación de sulfuros"}
<b>Patrón Detallar Estado</b>	
OBJETIVO:	detallar_estado(x)
CONDICIONES:	causa(y,x), efecto(x)
MEDIA:	plantilla-texto detallar_caracteristica(x,y)
<b>Patrón Graficar</b>	
OBJETIVO:	detallar_valor (x)
CONDICIONES:	x != null
MEDIA:	gráfico
CUERPO:	RANDON {makeLineGraph(t,z,y,w)            v makeBarGraph(t,z,y,w) }

La identificación de objetivos comunicativos para formar un discurso se ha basado en la Teoría de la Estructura Retórica [91], [92]. Dicha teoría entiende un discurso con un conjunto de relaciones retóricas entre partes de segmentos discursivos. La teoría proporciona un conjunto estándar prefijado del total de relaciones retóricas posibles. En la presente propuesta, se han seleccionado y adaptado parte de

dichas relaciones retóricas para identificar objetivos comunicativos aplicables a la presentación de información sobre el árbol de inferencia arrojado por Drools.

Tabla 7. Objetivos comunicativos

<b>Objetivo comunicativo</b>	<b>Descripción</b>	<b>Ejemplo</b>
<i>informar_titular (x)</i>	Informa el valor instanciado en el estado <i>x</i> arrojado por la inferencia del Sistema Experto. Ejemplo, si el Sistema Experto muestra como salida temperatura alta en una determinada franja, esta estructura discursiva generaría automáticamente un segmento de texto tipo titular.	"La temperatura interna estimada es de 30 °C para la franja 209"
<i>sin_información ()</i>	Informa que el Sistema Experto no cuenta con datos suficientes en el estado <i>x</i> para generar una inferencia.	"No se cuenta información suficiente para generar una conclusión"
<i>detallar_estado (x)</i>	Describe en forma detallada el estado arrojado por la inferencia del Sistema Experto. Su uso por lo general es recursivo, por lo que es difícil de colocar un ejemplo concreto.	
<i>detallar_valor (x)</i>	Detalla en el paso del tiempo el comportamiento de un determinado concepto <i>x</i> .	Valores de temperatura versus cobre recuperado. Se utilizan gráficos de línea o de barra. en dependencia del operador seleccionado.

En la propuesta se puede hacer distinción entre dos tipos de condiciones:

- Condiciones sobre el Sistema Experto, relacionadas con los hechos que componen el árbol de inferencia.
- Condiciones sobre el usuario, estas condiciones están relacionadas con las preferencias del modo de presentación (tipo de gráfico, etc.).

A continuación, se describe en pseudocódigo el planificador de la propuesta Figura 14. Este planificador maneja los operadores de presentación y los patrones de discurso.

```

Program: Planificador
Environment: OBJECTS (lista), CHARACTERISTICS (lista),
MATRIZ_VALOR (matriz) y COMMENTS (lista)
Algorithm:
If Objects[-1] == "No predicción"
  Activate Patrón Titular 2 (objetivo comunicativo =
  sin_información())
End If
If len(Objects)>=1
  obj= Objects[-1]
  Activate Patrón Titular 1 (objetivo comunicativo =
  informar_titular (obj))
  Objects = Objects.Remove(obj)
  While len(Objects)>=0
    Activate Patrón Detallar Estado (objetivo comunicativo =
    detallar_estado(Objects[i]))
    If Objects[i].HasSibling()==True
      Activate Patrón Enumera (Objetivo comunicativo =
      enumera(i))
    End If
  End While
End If
Activate Patrón Graficar (objetivo comunicativo =
RANDOM(detallar_valor (x)) Y detallar_valor (microorganismos))
End Algorithm

```

Figura 14. Pseudocódigo de la tarea Planificar.

Dentro de la tarea Planificar se hace mención al método HasSibling. La función de este método es buscar en el RDF/XML todos los elementos que se encuentran en la lista Objects y tienen el mismo padre que el elemento x.

# **CAPÍTULO V: GENERACIÓN DE EXPLICACIONES EN EL DOMINIO DE LA BIOLIXIVIACIÓN**

---

## **5.1. Resumen del capítulo**

Este capítulo describe el resultado de la evaluación de la propuesta. Para llevar a cabo la evaluación se ha considerado importante disponer de un dominio real con suficiente complejidad. Este dominio nos debe permitir realizar experimentos que nos faciliten obtener conclusiones sobre la validez y utilidad práctica de la solución planteada. Se ha escogido el dominio minero, el cual resulta adecuado por las siguientes razones:

- *Disponibilidad de medidas.* Se cuenta con medidas periódicas extraídas del sistema dinámico que posibilitan a los expertos en el dominio caracterizar y tomar decisiones.
- *Disponibilidad de un Sistema Experto.* El conocimiento de los expertos debe poder ser representado mediante un Sistema Experto, de forma tal que el razonamiento seguido por él sea análogo al realizado por un humano experto en el dominio.
- *Apoyo a la toma de decisiones.* El Sistema Experto desarrollado debe apoyar a la toma de decisiones según diferentes objetivos (mantener la temperatura interna, facilitar la existencia de un determinado microorganismo, etc.), con el objetivo de obtener mayor cantidad de mineral.

La evaluación se ha realizado siguiendo la estrategia que se resume en los siguientes puntos:

- Comprobar la generalidad de la propuesta. Se trata de verificar la aplicabilidad de la propuesta sobre dominios concretos, reales de complejidad significativa.
- Comprobar la corrección operativa. Consiste en verificar que la propuesta sirve de base para generar explicaciones correctas sobre el árbol de inferencia recorrido por el Sistema Experto antes de ofrecer una conclusión.
- Evaluar la utilidad práctica. Esta tarea se realiza con la participación de un experto en el dominio. El propósito de esta evaluación es valorar la adecuación de la explicación. También se requiere participación de personas con conocimientos mínimos del dominio para valorar la facilidad de uso de la herramienta.

En los próximos apartados se detallan la forma y resultados de cada uno de estos tres aspectos evaluados.

## **5.2. Generalidad del trabajo**

La generalidad de la propuesta se ha comprobado con ayuda del dominio minero (extracción de cobre), específicamente en el proceso de Biolixiviación que ha permitido experimentar con diferentes objetivos de gestión [93].

En este dominio se cuenta con un Sistema Experto compuesto por tres módulos:

- *Estimación de la temperatura interna*; el contenido de este módulo incluye una sección de información y una sección de predicción para la franja seleccionada.
- *Cinética de oxidación*; módulo que caracteriza la actividad oxidante de una franja.

- *Genes indicadores*; este módulo entrega información oportuna respecto a cómo están reaccionando los microorganismos frente a cuatro tipos de variaciones operacionales.

Como paso previo a la descripción del módulo de temperatura, escogido para probar la propuesta, se presenta un resumen del dominio en cuestión para facilitar el entendimiento.

### **5.2.1. Descripción del dominio**

El cobre se extrae por dos vías generales: la piro-metalurgia y la hidrometalurgia. La primera trata los sulfuros de cobre a través del chancado y molienda, luego un proceso de flotación, para pasar a la fundición y electro-refinación. Mientras que la hidrometalurgia trata principalmente los óxidos de cobre, que se encuentran más cercanos a la superficie, donde el mineral pasa por una etapa de lixiviación, extracción por solvente y electro-obtención obteniéndose cátodos de cobre de alta pureza. Los sulfuros de cobre también pueden ser tratados con la hidrometalurgia, pero en este proceso se tiene una muy lenta disolución, por lo que se requiere un catalizador de la reacción, lo que ha fomentado el estudio e implementación de la biotecnología utilizando bacterias para la extracción del cobre, en el proceso de Biolixiviación [94].

La Biolixiviación consiste en apilar ripio (material resultante del proceso de extracción de cobre por métodos tradicionales) y sembrar este ripio con bacterias (irrigación) capaces de oxidar azufre generando ácido sulfúrico que permite solubilizar el mineral. El ripio se apila por niveles que se denominan pisos y el material solubilizado se denomina unidades PLS [93] donde se toman muestras y se estudia su composición para verificar la concentración de cobre [95]. La Figura 15 ilustra este proceso.

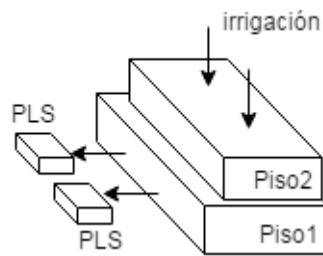


Figura 15 Esquema de una pila de Biolixiviación de cobre.

Las bacterias pueden acelerar el proceso de obtención del cobre. En una pila ellas pueden reproducirse de forma natural, creciendo unas u otras dependiendo de los diversos factores los cuales afectan su desarrollo. Estos factores son del tipo ambiental, biológico y fisicoquímicos. Uno de los factores ambientales es la temperatura. Es por eso que resulta de interés para los operarios e investigadores conocer la temperatura interna de cada una de las franjas para poder regularla y facilitar así el crecimiento de las bacterias y con esto, aumentar la obtención de cobre.

Los microorganismos que habitan en la pila son clasificados según el rango de temperatura en el que pueden sobrevivir. Los mesófilos tienen un rango óptimo entre 30-40°C, los termófilos moderados cercano a los 50°C, y los termófilos sobre los 65°C. Si la temperatura es menor a los 5°C se vuelven inactivas, mientras que si sobrepasa su rango de operación se mueren [96].

El Sistema Experto desarrollado es capaz de estimar la temperatura interna de las franjas de la pila conociendo los valores de temperatura del PLS, los tipos y cantidades de microorganismos que habitan en ella.

La Figura 16 muestra dos vistas de la pila en explotación. En la parte izquierda de la figura se puede observar una vista aérea de la pila mientras que en la parte derecha de la figura se muestra los pisos de la pila.



- Definir el alcance de la ontología.
- Evaluar la posibilidad de reutilización de otras ontologías y/o partes de ontología.
- Si existen ontologías que puedan ser reutilizadas habría que incluir el proceso para conseguir reutilizarlas, proceso que a su vez consta de diferentes fases y tareas a realizar.
- Enumerar los términos que aparecen en el dominio.
- Definir la taxonomía de conceptos. De arriba abajo, de abajo arriba o combinando las dos.
- Definir las propiedades.
- Definir las facetas.
- Definir las instancias.
- Comprobar las estructuras creadas.

Como se mencionó con antelación la evaluación de la propuesta fue realizada con el módulo de temperatura implementado por el Sistema Experto en el dominio de la Biolixiviación. De este módulo se pretende representar todo el conocimiento relacionado con los factores que pueden determinar la temperatura interna de una pila. Como restricción de la propuesta, se debe partir de los conceptos presentes en el modelo de objeto implementado por el Sistema Experto.

En una búsqueda preliminar realizada no se identificaron ontologías desarrolladas en este dominio, por lo que no se tuvo en cuenta la posibilidad de reutilización de ontologías existentes.

Se valoraron un total de 33 términos, véase la Tabla 8, de los cuales 26 fueron convertidos en clases y 7 en propiedades.

Tabla 8. Términos extraídos del dominio a representar

Estimación de la temperatura	Microorganismos	Tipo de Microorganismos
Arquea	Bacteria	Eukarya
Especie	A Thiooxidans	Leptospirillum SP
Género	Acidithiobacillus	Leptospirillum
Muestra	Tipo de Muestra	Dominio
Mesófilos	Termófilos	No Definido
Termófilo Moderado	Termotolerante	Electrolito Pobre
PLS	PLS Común	Refino
Muestra no Definida	Total de Microorganismos	Cantidad
Edad	Estado	Porcentaje
Rango de temperatura	Temperatura	Tolerancia

Se utilizó Protégè como herramienta para el diseño de la ontología. Al representar el conocimiento experto en una taxonomía de conceptos se obtuvo un total de 147 axiomas (teoremas que se declaran sobre relaciones que deben cumplir los elementos de la ontología) [97], los cuales permitirán en futuras versiones inferir conocimiento. Fueron identificadas 26 clases, 5 de ellas disyuntas y 2 equivalencias. Fueron declarados 58 axioma relacionados con las propiedades de los datos.

La Figura 18 muestra en el ámbito de la herramienta Protégè todos los conceptos identificados en una jerarquía de clases, puede verse aquellas clases que son equivalentes con otras y de una manera general la jerarquía de todas las clases identificadas del dominio. En la Figura 19 se muestran las propiedades de los objetos, para este caso sólo fueron identificadas dos relaciones binarias. Por último, la Figura 20 presenta las propiedades de datos identificadas en la ontología. Todas las propiedades identificadas están al mismo nivel no se presentan jerarquías en este caso.



Figura 18. Jerarquía de Clases.

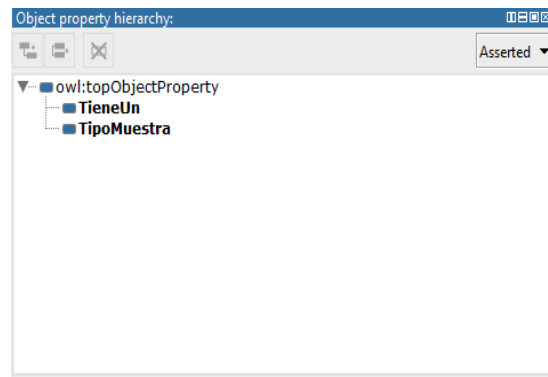


Figura 19. Propiedades de Objetos.

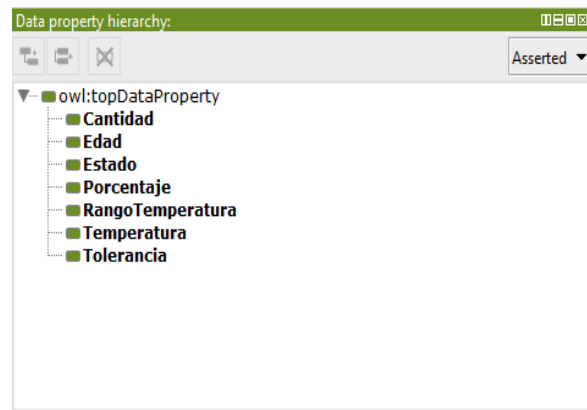


Figura 20. Propiedades de Datos.

Con el objetivo de ilustrar globalmente la representación del dominio, se presenta en la Figura 21 el grafo con todos los conceptos encontrados y las relaciones existentes entre cada uno de ellos.

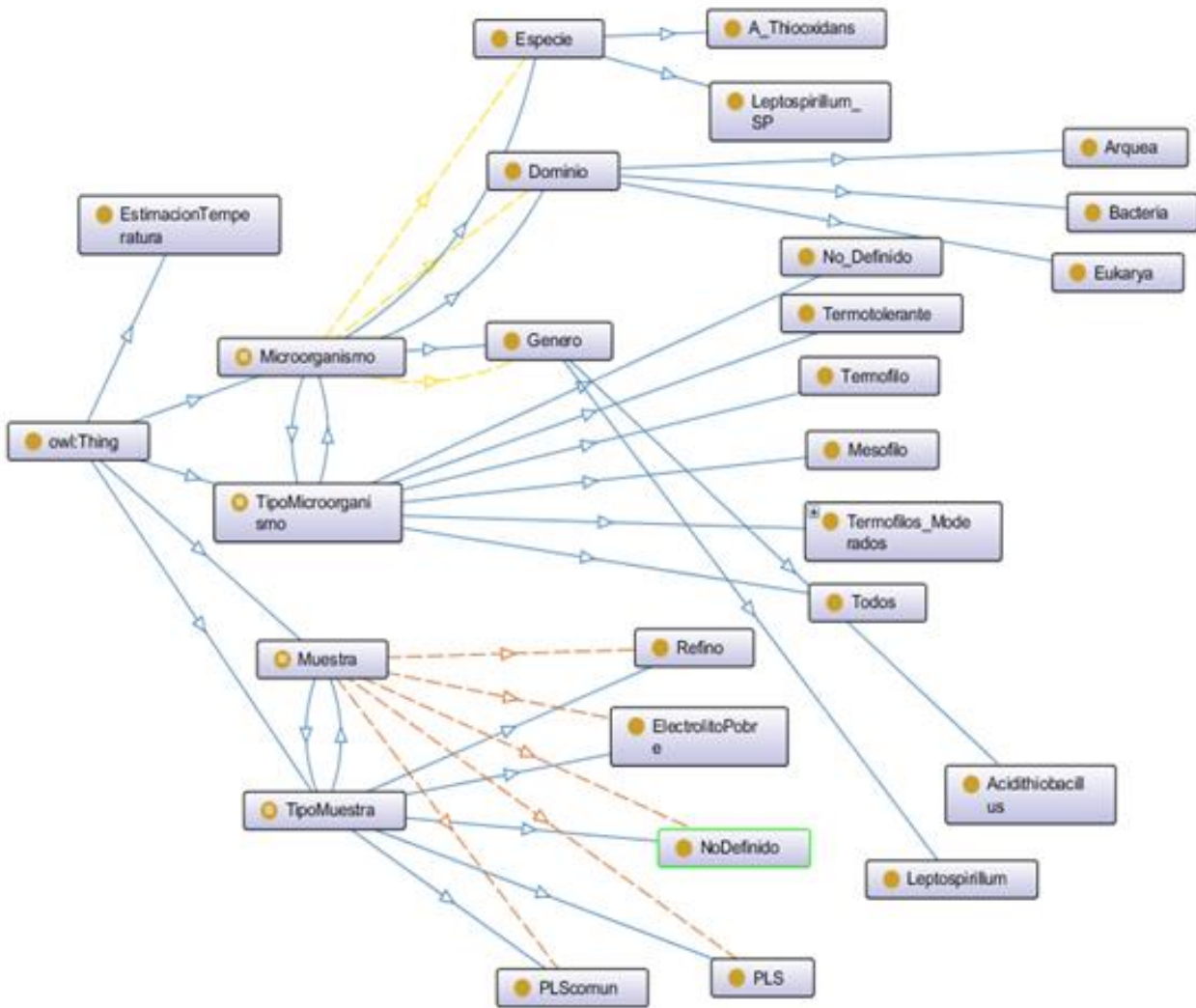


Figura 21. Gráfico que representa la Ontología en Protégè.

Después de haber representado todo el conocimiento del dominio en una ontología con ayuda de la herramienta Protégè se necesita para la entrada del método propuesto el RDF/XML que se obtiene tras el modelado. La Figura 22 presenta fragmentos de este archivo, en el anexo IV se puede encontrar el fichero completo.

En el RDF/XML que se muestra en la Figura 22 se puede apreciar las seis sesiones en que se organiza el RDF/XML además del encabezado y resumen del

fichero. El detalle de los que se representa en cada sesión fue explicado con profundidad en sesiones anteriores de este documento.

```

<?xml version="1.0"?>
<rdf:RDF xmlns="http://www.semanticweb.org/..."
  xml:base="http://www.semanticweb.org/..."
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <owl:Ontology rdf:about="http://www.semanticweb.org/..."/>

  //////////////////////////////////////
  // Object Properties
  //////////////////////////////////////

  <!-- http://www.semanticweb.org/...#TieneUn -->

  <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/...#TieneUn">
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Microorganismo"/>
    <rdfs:range
rdf:resource="http://www.semanticweb.org/...#Dominio"/>
    <rdfs:range
rdf:resource="http://www.semanticweb.org/...#Especie"/>
    <rdfs:range
rdf:resource="http://www.semanticweb.org/...#Genero"/>
  </owl:ObjectProperty>
  ...
  //////////////////////////////////////
  // Data properties
  //////////////////////////////////////

  <!-- http://www.semanticweb.org/...#Cantidad -->

  <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/...#Cantidad">
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#A_Thiooxidans"/>
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Acidithiobacillus"/>
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Arquea"/>
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Bacteria"/>
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Eukarya"/>
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Leptospirillum"/>
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Leptospirillum_SP"/>
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Mesofilo"/>
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#No_Definido"/>
    <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Termofilo"/>
    <rdfs:domain

```

```

rdf:resource="http://www.semanticweb.org/...#Termofilos_Moderados"/>
  <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Termotolerante"/>
  <rdfs:domain
rdf:resource="http://www.semanticweb.org/...#Todos"/>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#long"/>
  <rdfs:comment>La suma de</rdfs:comment>
  </owl:DatatypeProperty>
...
////////////////////////////////////
//Classes
////////////////////////////////////

  <!-- http://www.semanticweb.org/...#EstimacionTemperatura -->

  <owl:Class
rdf:about="http://www.semanticweb.org/...#EstimacionTemperatura">
  <rdfs:comment>La temperatura interna estimada es
de</rdfs:comment>
  <rdfs:comment>La temperatura interna estimada para la franja es
de</rdfs:comment>
  <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">La temperatura
interna no pudo ser estimada</rdfs:comment>
  <rdfs:comment>Las condiciones de operacion utilizadas para la
Biolixiviacion en la franja determinan una temperatura interna
de</rdfs:comment>
  <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">No se pudo
estimar la temperatura interna</rdfs:comment>
  </owl:Class>
...
////////////////////////////////////
//Individuals
////////////////////////////////////

  <!-- http://www.semanticweb.org/...#Individuo1 -->

  <owl:NamedIndividual rdf:about="http://www.semanticweb.org/...#A1">
  <rdf:type
rdf:resource="http://www.semanticweb.org/...#Termofilos_Moderados"/>
  </owl:NamedIndividual>

////////////////////////////////////
// General axioms
////////////////////////////////////

  <rdf:Description>
  <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#AllDisjointClasses"/>
  <owl:members rdf:parseType="Collection">
  <rdf:Description
rdf:about="http://www.semanticweb.org/...#Arquea"/>

```

```

        <rdf:Description
rdf:about="http://www.semanticweb.org/...#Bacteria"/>
        <rdf:Description
rdf:about="http://www.semanticweb.org/...#Eukarya"/>
        </owl:members>
    </rdf:Description>
    ...
</rdf:RDF>
<!-- Generated by the OWL API (version 4.2.5.20160517-0735)
https://github.com/owlcs/owlapi -->

```

Figura 22. Fragmento del RDF/XML obtenido del modelado.

### 5.2.2.1. Representación del dominio en el Sistema Experto.

El objetivo de los sistemas expertos es emular el razonamiento de expertos humanos en un dominio específico, como se vio en capítulos anteriores. Por esta razón los ingenieros de conocimiento después de varias reuniones realizadas con los expertos en el dominio en cuestión se han dado la tarea de extraer dicho conocimiento de manera que pueda ser procesado por un sistema.

El trabajo conjunto entre el ingeniero de conocimiento y los expertos en el dominio es fundamental, ya que el ingeniero de conocimiento no es un experto en el campo que intenta modelar, mientras que el experto en el dominio no tiene experiencia modelando su conocimiento (basado en la heurística), de forma que pueda ser representado de manera genérica en un sistema.

Después de varias sesiones dedicadas a la extracción y análisis del conocimiento fueron identificadas para el módulo de temperatura las siguientes reglas [93]:

- **SI** (suma de microorganismos < 10e5) **ENTONCES** no hay predicción (no se puede concluir sobre la temperatura interna)
- **SI** (suma de microorganismos >= 10e5) **Y** (temperatura de PLS >= 20) **Y** (% termófilos moderados (TM) >= 52%). **ENTONCES** (temperatura interna = (temperatura de PLS) +22)
- **SI** (temperatura de PLS >= 20) **Y** (% termófilos moderados (TM) < 52%) **ENTONCES** (temperatura interna = (temperatura de PLS) +10).

- **SI** (temperatura de PLS < 20) **ENTONCES** (temperatura interna = temperatura de PLS).

Para obtener el conocimiento representado en las reglas los expertos en ingeniería del conocimiento organizaron su trabajo siguiendo dos etapas:

- Prototipación: En esta etapa el ingeniero se familiariza con el vocabulario del dominio y obtiene el primer modelo de base de conocimiento. Los métodos utilizados para cumplir la etapa son las tareas familiares y las entrevistas no estructuradas [98].
- Refinamiento: En esta etapa se mejoran la calidad de las cotas asociada a los límites de las restricciones identificadas y se pueden definir mejor las condiciones identificadas con anterioridad. Para esta etapa los métodos utilizados son entrevistas estructuradas realizada a los expertos y tareas de procesamiento restringido [99].

El Sistema Experto fue dividido en dos grandes paquetes:

- Resources, en este paquete se encuentran todos los ficheros **.drl**<sup>5</sup> identificados, en estos ficheros es donde se gestionan las reglas que caracterizan al Sistema Experto.
- Java, este gran paquete a su vez fue dividido en 3 subpaquetes. El primero de esos paquetes agrupa toda la lógica del sistema (logic), el segundo agrupa cada una de las clases que gestionan las sesiones de Drools (control) y por último se cuenta con un paquete (enumerator) que agrupa todos los enumeradores que fueron creados para solucionar el problema.

La Figura 23, muestra el modelo de objeto utilizado para la implementación del módulo de temperatura en el Sistema Experto. Es importante destacar que cada uno de los objetos identificados en este paso deben estar contenido como conceptos en la ontología de dominio.

---

<sup>5</sup> Lenguaje de reglas en Drools, utilizado para especificar las reglas.

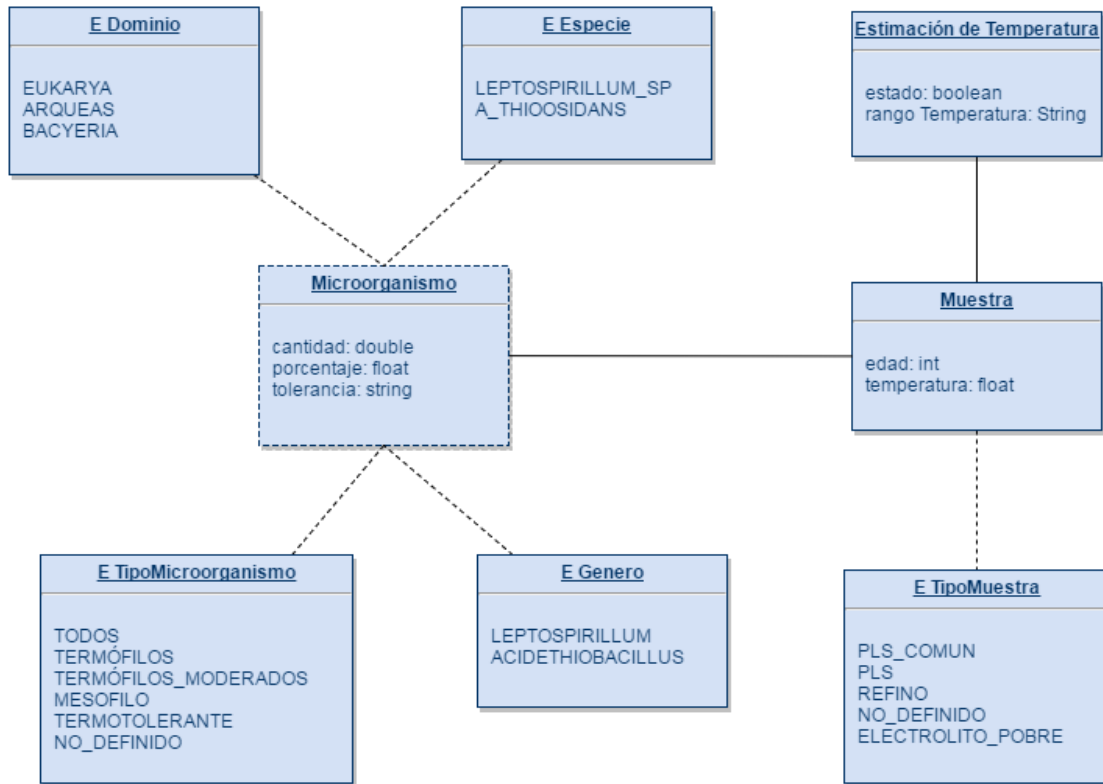


Figura 23. Diagrama de objetos del módulo de temperatura utilizado por el Sistema Experto.

Los objetos cuyo nombre comienzan con la letra **E** representan enumeradores. La propuesta establece que estos enumeradores sean tratados como conceptos al representar el conocimiento que engloban en la ontología. Por otra parte, los atributos serán referido en el comentario de cada concepto en dependencia de la relevancia para la explicación.

Una vez que se encuentre listo el sistema para su explotación se solicita mediante el registro de auditoría el archivo FileLogger. Este archivo junto con el RDF/XML son los que necesita la propuesta para proceder con la explicación. En la Figura 24, se muestra un fragmento del archivo generado para la franja 309 de la pila de Biolixiviación, en el anexo V se puede encontrar el archivo completo. No fue escogida esta franja por ninguna razón en particular.

```

<object-stream>
<org.drools.core.audit.WorkingMemoryLog>
  <version>6.1</version>
  <events>
    <org.drools.core.audit.event.ObjectLogEvent>
      <type>1</type>
      <factId>393</factId>
<objectToString>Microorganismo.cantidad[value=39428]</objectToString>
</org.drools.core.audit.event.ObjectLogEvent>
<org.drools.core.audit.event.ObjectLogEvent>
  <type>1</type>
  <factId>394</factId>
  <objectToString>Muestra.temperatura[value=17,5]</objectToString>
</org.drools.core.audit.event.ObjectLogEvent>
  ...
  <org.drools.core.audit.event.ObjectLogEvent>
    <type>1</type>
    <factId>396</factId>
<objectToString>EstimacionTemperatura.estado[value=false]</objectToString>
</org.drools.core.audit.event.ObjectLogEvent>
<org.drools.core.audit.event.ActivationLogEvent>
  <type>4</type>
  <activationId>Si la suma de microorganismos es menor que 10e5[393,
396]</activationId>
  <rule>Si la suma de microorganismos es menor que 10e5</rule>
  <declarations>$tempEstimada=false</declarations>
  <factHandleIds>396,393</factHandleIds>
</org.drools.core.audit.event.ActivationLogEvent>
<org.drools.core.audit.event.ActivationLogEvent>
  <type>6</type>
  <activationId>Si la suma de microorganismos es menor que 10e5[393,
396]</activationId>
  <rule>Si la suma de microorganismos es menor que 10e5</rule>
  <declarations>$tempEstimada=false</declarations>
  <factHandleIds>396,393</factHandleIds>
</org.drools.core.audit.event.ActivationLogEvent>
<org.drools.core.audit.event.ObjectLogEvent>
  <type>2</type>
  <factId>396</factId>
  <objectToString>EstimacionTemperatura.estado["No se puede estimar"]
</objectToString>
</org.drools.core.audit.event.ObjectLogEvent>
<org.drools.core.audit.event.ActivationLogEvent>
  <type>7</type>
  <activationId>Si la suma de microorganismos es menor que 10e5[393,
396]</activationId>
  <rule>Si la suma de microorganismos es menor que 10e5</rule>
  <declarations>EstimacionTemperatura.estado="No se puede
estimar"</declarations>
  <factHandleIds>396,393</factHandleIds>
</org.drools.core.audit.event.ActivationLogEvent>
<org.drools.core.audit.event.ObjectLogEvent>
  <type>3</type>
  <factId>394</factId>
  <objectToString>Muestra.temperatura[value=0]</objectToString>
</org.drools.core.audit.event.ObjectLogEvent>
  ...

```

```
</events>  
<engine>PHREAK</engine>  
</org.drools.core.audit.WorkingMemoryLog>  
</object-stream>
```

Figura 24. Fichero Log obtenido del Drools como resultado de la inferencia

Hasta el momento, fueron descritos todos los datos necesarios para que el modelo propuesto pueda proporcionar de manera automática, explicaciones basadas en el árbol de inferencia ofrecido por Drools y la ontología desarrollada en el dominio específico. A continuación, se muestra en detalle cada una de las tareas que componen el método.

#### **5.2.2.2. Modelo de Presentación**

Para generar la explicación sobre el comportamiento del Sistema Experto en el dominio minero, esta propuesta sigue una estrategia, donde primero presenta la información más agregada, seguida de un discurso que aporta los detalles. La explicación contempla una parte de texto y una secuencia gráfica que resume los datos que se cuentan. La Figura 25 ilustra la estructura general del discurso.

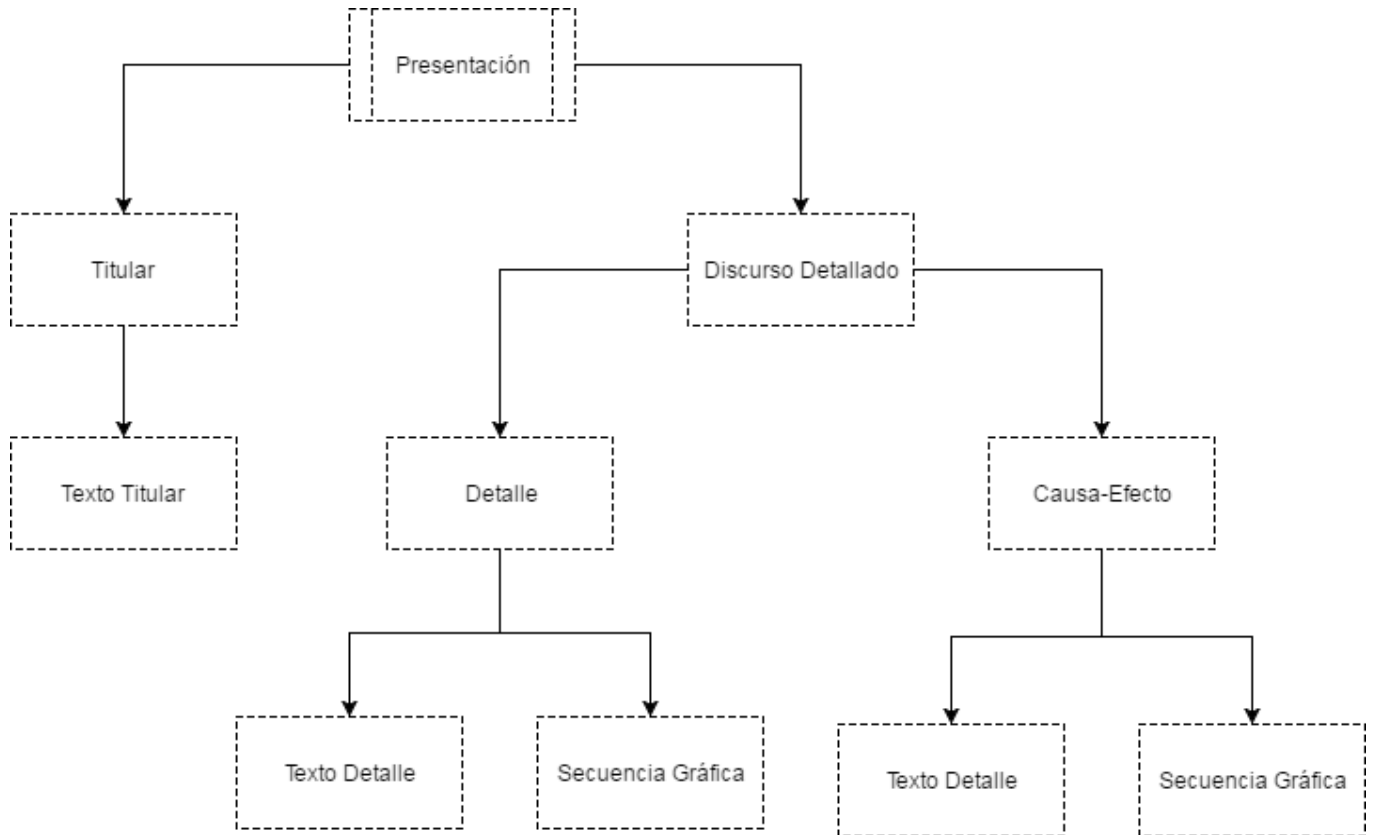


Figura 25 Esquema de planificación de la presentación

Dicha estructura se implementa en la propuesta con los patrones de discurso que utiliza el planificador. Por ejemplo, para expresar la parte de discurso correspondiente al discurso detallado, según la Figura 25 se utiliza el patrón siguiente:

**OBJETIVO:** *titular(x)*  
**CONDICIONES:** *causa(y,x), efecto(x)*  
**DISCURSO:** *detallar\_estado(y),  
 detallar\_caracteristica(x,y)*

La propuesta usa plantillas de frases que pueden ser clasificadas en cuatro tipos: titular, detalle, causa y efecto. Las plantillas de texto de tipo titular se usan para generar frases sobre la conclusión arrojada por el Sistema Experto, la cual contiene al más alto nivel de generalidad, como por ejemplo, la temperatura interna en determinada franja.

Las plantillas de texto tipo detalle son utilizadas para complementar la información del hecho más general reflejada en el titular, aportando ciertos detalles informativos del hecho de interés. Por último, las plantillas de tipo, causa o efecto reflejan información correspondiente a estados causas o efectos, relacionados con el estado más agregado del sistema, también son las encargadas de garantizar la relación causa- efecto entre dos hechos.

### **5.3. Resultados y discusión**

Para evaluar la operación correcta de la propuesta, el método utilizado fue estudio de caso. A pesar de que el estudio de caso no da lugar a relaciones causales como ocurre con los experimentos, éste permite comprender más en profundidad fenómenos en un contexto real. Este método tiene la capacidad de proporcionar resultados de investigación a partir de proyectos del mundo real como es el caso de la Biolixiviación de cobre en una pila.

Existen un conjunto de elementos necesarios para planificar un estudio de caso [100], uno de estos elementos es la recolección de datos. Para esta investigación se tiene un conjunto de 86 franjas que corresponde al total de datos con que se cuenta en la pila de Biolixiviación. Estas franjas se encuentran distribuidas en 4 pisos y además, pueden ser divididas en franjas operativas y no operativas quedando los datos clasificados como se muestra en la Figura 26.

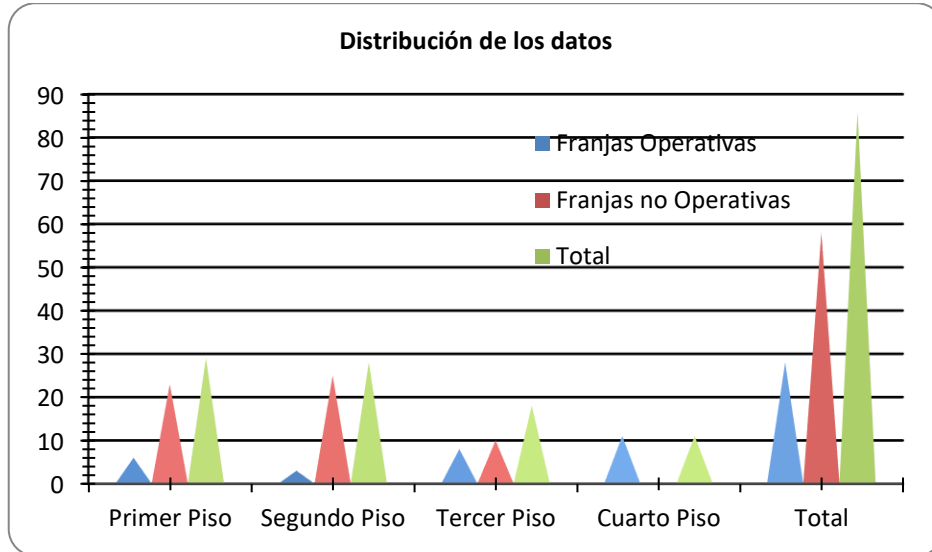


Figura 26 Distribución de los datos

Del total de los datos con que se contaban fueron seleccionados un conjunto de 19 casos. Estos 19 casos fueron identificados como típicos.

### 5.3.1. Selección de los casos Típicos

La pila a la que se le está realizando todo el análisis cuenta con 4 pisos como se mencionó anteriormente. Se considera que cada uno de los pisos pueden tener un comportamiento similar pero no tiene por qué ocurrir lo mismo en toda la pila ya que la distribución física, la cantidad de material etc., pudiera estar ocasionando alguna alteración en su funcionamiento. Basados en esto se puede identificar que para cada piso vamos a contar con al menos un caso típico.

El segundo problema que se encontró al identificar casos típicos fue que por cada piso las franjas pueden estar siendo operativas o no operativas. Ya que el tratamiento que reciben cada una de las clasificaciones de las franjas es distinta y son tenidas en cuenta para identificar los casos típicos.

El último aspecto que se tuvo en cuenta para identificar casos típicos fue el resultado que arroja el Sistema Experto según las reglas que fueron implementadas. Como se mencionó con anterioridad, el Sistema Experto cuenta

con cuatro reglas implementadas para el módulo de temperatura. Cada una de estas reglas se tienen en cuenta para la identificación de los casos típicos.

La Figura 27 muestra un diagrama donde puede identificarse con mayor facilidad lo antes explicado.

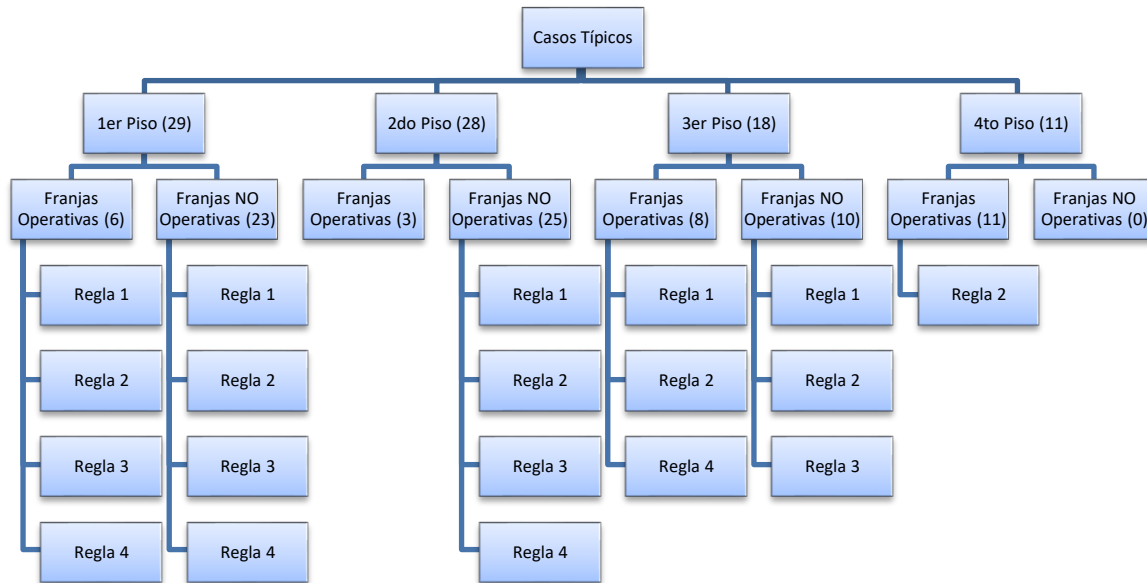


Figura 27. Casos Típicos Identificados

### 5.3.2. Evaluación de la utilidad práctica

En este apartado se describe la evaluación de la utilidad práctica del trabajo. Ésta consiste en evaluar con ayuda de un experto las explicaciones ofrecidas por el sistema, para ellos se utiliza la técnica de post-edición [10], [101] teniendo en cuenta algunas métricas identificadas como de interés para la evaluación.

#### 5.3.2.1. Técnica post-edición

La evaluación de la utilidad práctica se realiza con ayuda de un experto, se trata de aplicar la técnica de post-edición usada en trabajos como [102] [103]. Esta técnica consiste en que el experto usa las explicaciones generadas por el sistema

y los datos utilizados que se manejaron para llegar a ellas con el fin de obtener la explicación y en base a eso el experto se crea su propia versión del texto explicativo. La cantidad de cambios que realiza el experto sobre la explicación original cuantifica el grado de satisfacción que se tiene en el texto generado.

Para esta evaluación las explicaciones generadas automáticamente son entregadas al experto en conjunto con los datos para que elimine o sustituya unas palabras que forman parte del discurso por otras, manteniendo en lo posible el esquema general de la explicación generada automáticamente.

Con el fin de realizar esta prueba, el experto cuenta con un conjunto de  $K$  textos generados automáticamente y el experto genera un conjunto de  $K$  textos modificados. La evaluación post-edición contempla la obtención de dos medidas:

- *Validez del texto original.* La validez del texto original se mide con el parámetro  $V$  que toma valores en forma de porcentaje de 0 a 100, donde el valor 100 indica que el texto generado resulta válido completamente. Para calcular este valor se utiliza la siguiente fórmula:

$$V = \frac{100(N - E)}{N}$$

donde  $N$  es el número de caracteres del texto original y  $E$  es el número de caracteres correspondientes a palabras que se han eliminado del texto original.

- *Presencia en el texto modificado.* Se mide también la cantidad de texto original presente en el texto modificado. Se mide con el parámetro  $P$  que toma valores de 0 a 100, donde el valor 100 indica que el texto original se ha mantenido completamente. Se aplica la fórmula siguiente:

$$P = \frac{100(M - V)}{M}$$

donde  $M$  es el número de caracteres del texto modificado y  $V$  es el número de caracteres correspondientes a palabras que son nuevas en el texto modificado.

Para la realización de la evaluación se escogió la explicación ofrecida por el sistema para la franja 403 de la muestra obtenida el 15 de enero del 2015. La razón por la cual se escogió este conjunto de datos para generar la explicación es que en ella se encuentran involucradas la regla que tiene en cuenta la mayor cantidad de variables con datos suficientes para que el árbol de inferencia se torne complejo.

La Figura 28 muestra un ejemplo de texto original utilizado en la evaluación. Las palabras subrayadas corresponden a palabras eliminadas en el texto modificado. La medida  $V$  para este ejemplo es de 87,9%

**Las condiciones ~~de operación utilizadas~~ para la Biolixiviación en la franja determinan una temperatura interna de 52°C**

La temperatura interna obtenida se debe a que la suma de los microorganismos ~~es mayor a 10e5, para este caso es de~~ 3057195. Además, la temperatura del PLS supera los 20 °C, alcanzando el valor de 29,8. La temperatura interna obtenida se debe además de los factores descrito anteriormente a que en la franja habitan más de 52% de microorganismos Termo\_tolerantes

Figura 28. Ejemplo de texto original con palabras modificadas  
( $V = 100(400 - 45)/400$ )

La figura 29 por su parte, muestra el ejemplo utilizado para la evaluación de la segunda parte de la técnica. Las palabras subrayadas en esta ocasión corresponden a palabras añadidas con respecto al texto original. La medida  $P$  es este ejemplo tiene un valor de 86,3%

**Las condiciones operacionales para la Biolixiviación en la franja determinan una temperatura interna de 52°C**

La temperatura interna obtenida se debe a que la suma de los microorganismos supera el valor de 10e5, específicamente es de 3057195. Además, la temperatura del PLS supera los 20 °C, alcanzando el valor de 29,8 °C. La temperatura interna obtenida se debe, además de los factores descrito anteriormente, a que en la franja habitan más de 52% de microorganismos Termo\_tolerantes.

Figura 29. Ejemplo de texto modificado con palabras añadidas  
( $P = 100(410 - 47)/410$ )

La aplicación de esta evaluación generó los valores medios de  $V = 87.9\%$  (desviación estándar de 147 caracteres) y  $P = 83,3\%$  (desviación estándar de 198 caracteres). La primera medida, muestra un alto valor de validez de los textos generados. La segunda medida presenta también un alto valor, pero más bajo que el anterior.

### 5.3.2.2. Contraste con el experto

Otra herramienta utilizada para la evaluación de la utilidad práctica del trabajo fue la encuesta, ya que esta nos permite recolectar datos sobre actitudes, creencias y opiniones de los individuos de una población. Además, las encuestas permiten hacer análisis estadísticos fácilmente [104].

En este trabajo se aplicaron dos encuestas a dos poblaciones distintas. El primer grupo de encuestados está caracterizado por ser expertos en el tema de la Biolixiviación. El objetivo de la encuesta aplicada a este primer grupo es determinar en qué porcentaje la explicación cubre los tres parámetros que se describen a continuación.

- Cobertura: esta métrica se encarga de medir el grado de soporte (cobertura) ofrecida por la explicación. Se le pide al experto que plasme su opinión sobre la presencia de todas las variables que deben estar según los datos que maneja el Sistema Experto.
- Claridad: esta métrica es la encargada de recoger la en porcentaje (%) de claridad, nitidez y fluidez de la explicación.
- Coherencia: esta métrica registrará el grado de coherencia de la explicación con los fenómenos representados por el Sistema Experto.

Para la selección de la muestra representativa se utilizó un muestreo no aleatorio. En relación al cuestionario, el tipo utilizado es estructurado ya que presenta

preguntas cerradas y otorga muy poca libertad. En el anexo I se podrá encontrar el cuestionario aplicado [104].

No fue fácil encontrar expertos en el tema a los cuales se les pudiera aplicar la encuesta. Por esta razón el número de la muestra es de 5 profesionales expertos en el área. Dentro de la Universidad Católica del Norte se cuenta aproximadamente una población de 9 profesionales que se pudiera considerar expertos en el proceso en cuestión, bajo esta situación se considera que la muestra es representativa de la población con la que se cuenta.

En el anexo II se puede encontrar el modelo de encuesta aplicado. Como se puede apreciar se puso a consideración de los encuestados 4 textos de los cuales se les pidió evaluar los criterios antes explicados (cobertura, claridad y coherencia). Los resultados arrojados se encuentran tabulados en la Figura 30

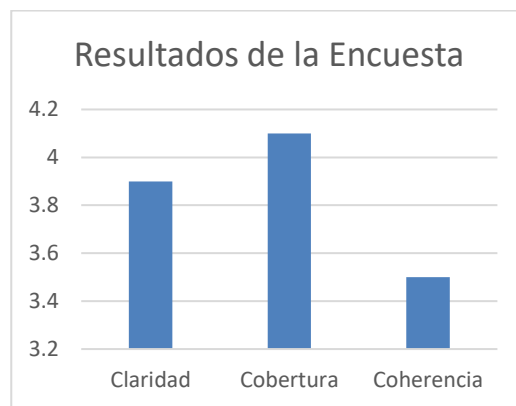


Figura 30 Tabulación de la Encuesta

Como puede apreciarse la métrica con valor más bajo fue la coherencia. Este resultado nos incita a continuar con la investigación adicionándole técnicas de procesamiento del lenguaje natural (PLN).

El segundo tipo de población a la cual se le aplicó la encuesta está caracterizado por personas con conocimientos medios y avanzados en el área de la informática esta población está compuesta por estudiantes de primer año de todas las ingenierías, egresados de las carreras Ingeniería Civil y Ejecución en Informática.

El total de la muestra es de 51 persona. El objetivo de esta encuesta es determinar en qué porcentaje la explicación cubre los parámetros que se describen a continuación:

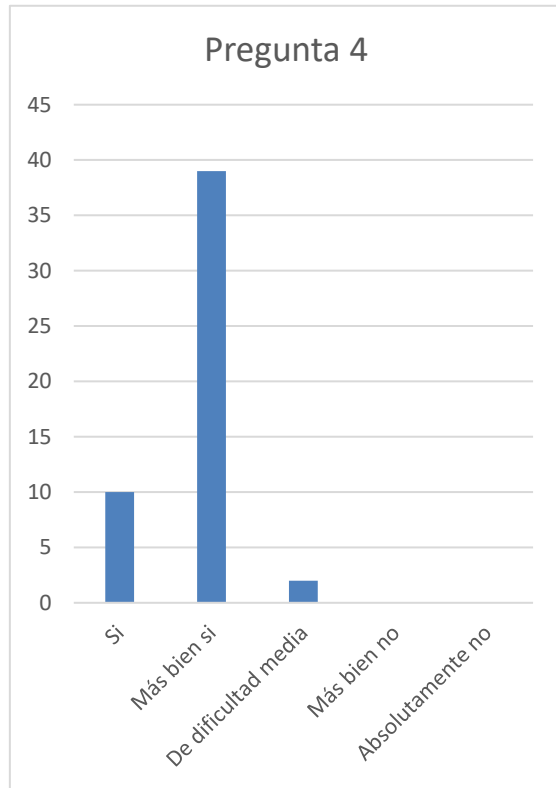
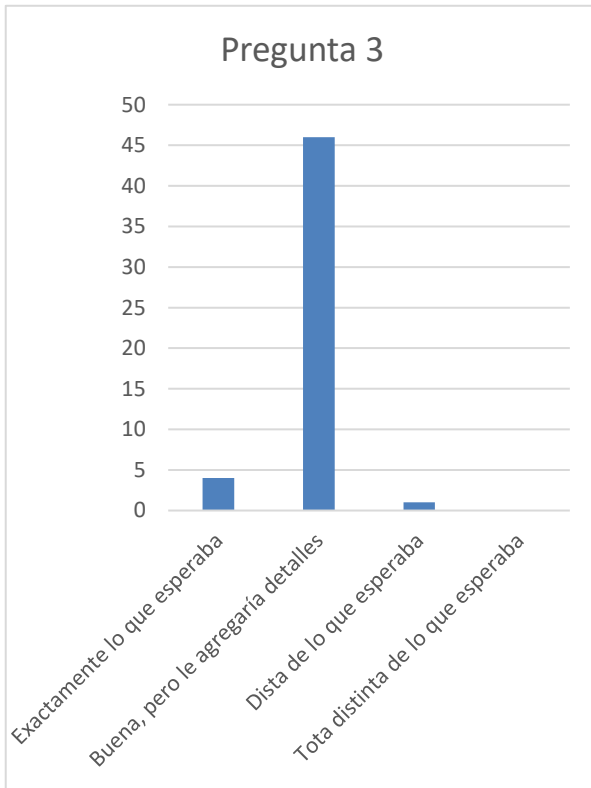
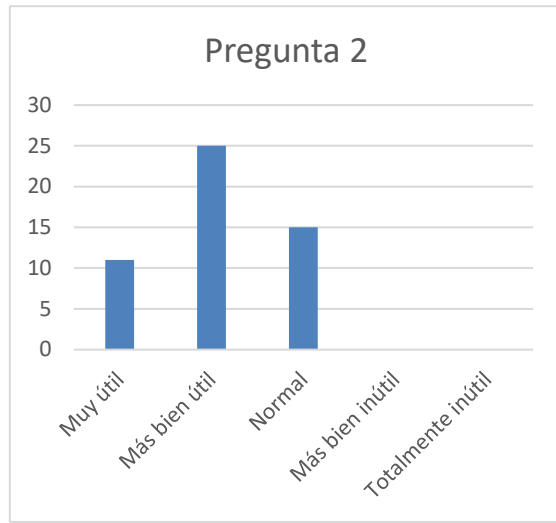
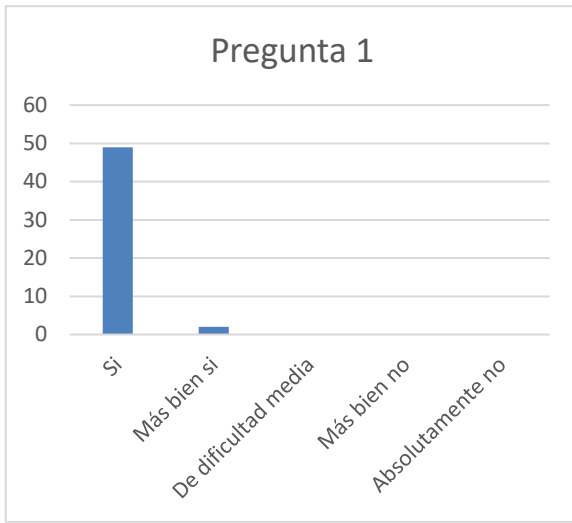
- Facilidad de uso del prototipo.
- Disponibilidad de la documentación.
- Facilidad para introducir las plantillas de texto.
- Facilidad de para introducir nuevos conceptos.
- Medir la inteligencia del algoritmo (que la explicación otorgada parezca como si la hubiese dado un humano).

Para medir estos objetivos se plantearon 5 preguntas las cuales se muestran a continuación:

1. ¿La interfaz del prototipo para la generación de explicaciones automáticas es fácil de usar?
2. La documentación que acompaña al prototipo es
3. La explicación que muestra el prototipo es
4. ¿La forma de insertar las plantillas de texto es fácil de usar?
5. La explicación brindada por el prototipo de generación automática de explicaciones

A continuación, se muestran los valores tabulados ofrecidos para cada pregunta por los encuestados (ver Figura 31).

CAPÍTULO V: GENERACIÓN DE EXPLICACIONES EN EL DOMINIO DE LA BIOLIXIVIACIÓN



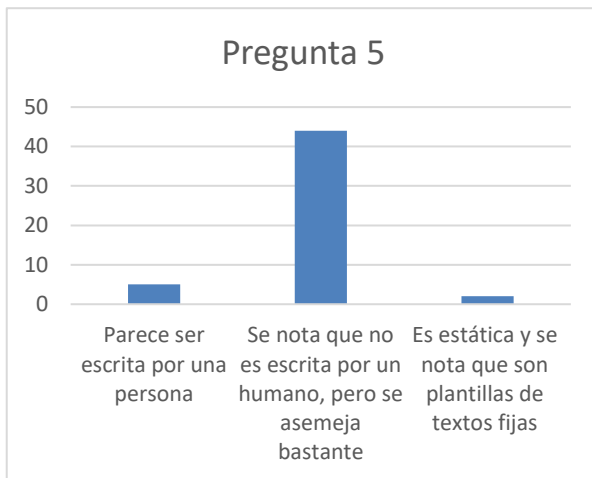


Figura 31 Tabulación de los resultados de cada una de las pregunta realizada a los expertos.

Los resultados obtenidos para la pregunta 1 indican que la interfaz prototipo utilizada para presentar la investigación es de fácil uso, ya que el 96% de los entrevistados marcaron la respuesta **si** como la más adecuada.

Para la pregunta número 2 se tiene trabajo que realizar, según los encuestados el documento con que se cuenta para explicar en detalle el dominio y entender, así como elaborar las plantillas de texto adecuadas es útil pero el 78% opina que puede mejorarse para llegar a un entendimiento total.

La pregunta 3 que recoge la expectativa de los usuarios, el 90% de ellos indican que la explicación es buena pero que le agregarían unos detalles. El 7% opinan que la explicación no es extraordinaria, pero ninguno de los encuestados dio evaluación negativa de los textos que leyeron.

Por otra parte, la pregunta 5, estrechamente relacionada con la pregunta 3, el 88% de los encuestados indican que no pueden asegurar que la explicación sea escrita por un humano, pero se asemeja bastante.

Por último, la pregunta 4 indica que las plantillas de texto una vez entendido como funciona el proceso es fácil de usar.

En base a los resultados que se presentan con las encuestas se puede concluir que el algoritmo planteado es capaz de generar explicaciones cercanas a como lo haría un ser humano, que la interfaz para el prototipo es amigable y la manera establecida para modificar o agregar plantillas de frases no presenta complicación.

# **CAPÍTULO VI: CONCLUSIONES**

---

En este capítulo se resume el trabajo realizado en relación con los objetivos planteados, se destacan los aportes principales y se indican posibles líneas futuras de investigación.

## **6.1. Resumen**

El objetivo principal del presente trabajo se ha cumplido a cabalidad, este ha sido la implementación de un algoritmo para la construcción automática de explicaciones en un contexto de explotación de un sistema dinámico complejo, facilitando la comprensión del proceso de inferencia y la conclusión a la cual llega un Sistema Experto. Este algoritmo presenta las explicaciones de forma comprensible y cercana a como lo hiciera un experto humano, apoyando de esta manera el proceso de toma de decisiones.

De acuerdo con los objetivos planteados en este trabajo, se realizó en primer lugar un análisis de la generación de explicaciones en sistemas dinámicos. Se analizaron las necesidades profundizando en el problema para el caso de sistemas dinámicos supervisados por equipos humanos responsables de la actuación según objetivos prefijados.

El trabajo incluyó la revisión del estado del arte sobre los temas relacionados con el objetivo general. En concreto, se revisaron los avances recientes dentro del campo de la Inteligencia Artificial en las áreas de modelado de sistemas dinámico, de generación automática de explicaciones y la forma de presentar esa información.

Tras dicha revisión, se diseñó una solución general para generar y presentar explicaciones que combina la representación del conocimiento de sistema dinámico en una ontología de dominio y describir el proceso de inferencia por el cual transita el Sistema Experto para entregar una conclusión. La solución propuesta: relata información relevante para el Sistema Experto y construye

presentaciones multimedia (en forma texto) que explican de forma resumida y en un lenguaje comprensible los datos considerados para construir la explicación.

El trabajo se validó con el fin de comprobar su aplicabilidad, corrección y utilidad práctica. Para ello, fue basado en un Sistema Experto desarrollado en el dominio de la Biolixiviación, que junto a la representación ontológica de dicho dominio permitieron validar los algoritmos.

## **6.2. Principales aportes**

Como aporte principal, el presente trabajo contribuye con un algoritmo para la generación automática de explicaciones que describen el comportamiento de un sistema dinámico gestionado de acuerdo con objetivos prefijados. Ligado al aporte principal se pueden nombrar otras contribuciones como:

- Representación ontológica del proceso de Biolixiviación del Cobre. Esta investigación ofrece una ontología que describe unos de los procesos de la Biolixiviación, específicamente la estimación de temperatura interna en una pila. Esta ontología brinda una base en ámbito de la representación del conocimiento en esta área.
- En este trabajo se crea una conexión directa a los estados intermedios del árbol de inferencia de un Sistema Experto, para usarlo como entrada al modelo de presentación, junto con el estado del sistema expresado en valores de variables (que instancian características de los conceptos de la ontología).
- Descripción minuciosa de los archivos XML/RDF generados por Protègè y WorkingMemoryLogger construido por el Drools, cuando realiza sus inferencias. Se ofrece una forma creativa de vincular estos dos elementos en aras de obtener una descripción automática cercana a como lo hiciera un humano.
- Se implementan Patrones de Discurso que aportan como contribución una solución eficiente para la combinación de formas de presentación (texto, gráficos, imágenes, etc), haciendo uso de los objetivos comunicativos con

estrategias de presentación. Este tipo de soluciones se ha aplicado en prototipos sencillos en otros problemas. Como novedad principal, en este trabajo se desarrolló una solución que hace uso de un dominio complejo junto a mecanismos de presentación particulares (texto en lenguaje natural).

- El algoritmo de presentación incluyó un estilo propio derivado del enfoque periodístico para concebir el tipo de presentaciones. Se trata de un estilo utilizado comúnmente y, por tanto, más asequible a un amplio sector de usuarios. Dicho estilo permitió generar formas de presentación intuitivas en línea con los objetivos de este trabajo.
- La construcción del algoritmo de presentación para un dominio concretos permitió identificar estrategias de presentación simples, representadas en forma de patrones de discursos. Esta estrategia puede ser generalizada y por tanto reutilizables para dominios diferentes donde es necesario generar explicaciones desde una ontología y el Drools.

En el marco de este trabajo de tesis fueron presentado los avances parciales en diferentes eventos científicos, obteniendo una retroalimentación satisfactoria que apoyo el buen término de la tesis. Los trabajos publicados con el material que acá se muestra son:

- Víctor Flores, Yahima Hadfeg, Claudio Meneses. Generating Natural Language Explanations from Knowledge-Based Systems Results, Using Ontology and Discourses Patterns. 2017 International Joint Conference on Rough (IJCRS), Olsztyn, Poland
- Victor Flores, Yahima Hadfeg. A method for Automatic generation of explanations for cuper Bioleaching process in Chile. Congreso Internacional en Innovación Tecnologías Informática (CIISTI) 2016, Zacatecas, Mexico
- Yahima Hadfeg, Víctor Flores, Nelson Morales. Generación Automática de Explicaciones desde una Base de Regla y una Ontología de Dominio en un Sistema Dinámico. III Congreso Internacional de Ingeniería Informática y Sistemas de Información. CIISI 2016, La Habana, Cuba

- Yahima Hadfeg, Víctor Flores, Claudio Meneses. "A method for automatic generation of explanations from a Rule-based Expert System and Ontology". 5th International Conference on SPI. CIMPS 2016. Aguascaliente, México,
- Yahima Hadfeg, Víctor Flores, Claudio Meneses. "Un método para la Generación Automática de Explicaciones de resultados de un Sistema Experto, centrado en la Biolixiviación de cobre". 7th International Conference on Computing and Informatics in Northern Chile. INFONOR 2016, Chile.

### **6.3. Limitaciones y trabajos futuros**

Existen diferentes recomendaciones que pudieran aportar a la investigación realizada, dictando el camino a seguir para futuros trabajos. Como punto de partida y que suponen una evolución a partir de sus limitaciones se pueden considerar los siguientes:

- Generación de texto mediante técnicas de procesamiento de lenguaje natural. En la presente propuesta se hace uso de un método de generación de texto basado en plantillas, lo cual aporta eficiencia en dicho proceso. No obstante, con el fin de mejorar en flexibilidad, es posible incluir técnicas de generación de lenguaje natural.
- Modelos alternativos de coordinación texto y gráficos. Se trata de explorar el uso de formas de representación que puede aportar mayor flexibilidad en la coordinación de formas de presentación, teniendo en cuenta aspectos de eficiencia en modelos complejos.
- Extensión de estrategias de discurso. El objetivo es explorar conjuntos de estrategias de discurso, en ambientes multimedia, que sean reutilizables para un mayor conjunto de problemas donde es útil generar explicaciones.
- Obtención automática de la ontología a partir del modelo de objeto para la implementación del Sistema Experto o viceversa. El objetivo es establecer mecanismos que permitan obtener la base de la ontología una vez que se

tiene el Sistema Experto implementado o partir con el modelado de la ontología para la construcción del mencionada Sistema Experto. Esto garantizaría una relación natural entre ambas modelos facilitando así la navegación para obtener la explicación.

## REFERENCIAS BIBLIOGRÁFICAS

---

- [1] R. Isermann and M. Munchhof, *Identification of Dynamic Systems: An Introduction with Applications*. New York: Springer, 2010.
- [2] P. W. Sauer, M. A. Pai, and J. H. (Joe H. . Chow, *Power system dynamics and stability: with synchrophasor measurement and power system toolbox*, Wiley. 2017.
- [3] S. Marcus, *Automating Knowledge Acquisition for Expert Systems*. Springer US, 1988.
- [4] J. David, J. Krivine, and R. Simmons, *Second generation expert systems: A step Forward in Knowledge Engineering*. Springer Science & Business Media, 2012.
- [5] G. Sagerer and H. Niemann, *Semantic networks for understanding scenes* . Springer Science & Business Media, 2013.
- [6] A. Garrido, *Logical Foundations of Artificial Intelligence*, vol. 1, no. 2. EduSoft Pub, 2010.
- [7] D. J. Clancy and K. Benjamin, "Model Decomposition and Simulation: A component based qualitative simulation algorithm \*," in *Fourteenth National Conference on Artificial Intelligence*, 1997, p. 7.
- [8] D. J. Patterson, L. Liao, D. Fox, and H. Kautz, "Inferring High-Level Behavior from Low-Level Sensors," in *International Conference on Ubiquitous Computing*, 2003, pp. 73–89.
- [9] F. E. Cellier, A. Nebot, F. Mugica, and A. De Albornoz, "Combined Qualitative/Quantitative Simulation Models of Continuous-Time Processes Using Fuzzy Inductive Reasoning Techniques," *Int. J. Gen. Syst.*, vol. 24, no. 1–2, pp. 95–116, Jan. 1996.

- [10] V. Flores, "Generación Automática de Resúmenes de Comportamiento de Sistemas Dinámicos Mediante Modelos de Conocimiento del Dominio.," Universidad Politécnica de Madrid, Madrid, 2010.
- [11] M. Molina and V. Flores, "Generating multimedia presentations that summarize the behavior of dynamic systems using a model-based approach," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 2759–2770, 2012.
- [12] R. Shannon, "Introduction to the Art and Science of Simulation," in *Proceedings of the 30th conference on Winter simulation*, 1998, pp. 7–14.
- [13] T. J. M. Bench-Capon, *Knowledge representation: an approach to artificial intelligence*. Singapore: Academic Press INC, 1990.
- [14] M. Crisosto and F. Sanchis, "Gestión del conocimiento: representación y métricas. Utilización del método Dacum," *Rev. Ing. Ind.*, vol. 1, pp. 5–14, 2002.
- [15] D. Hume, V. Sanfélix, C. Ors Marqués, and B. Stroud, *Investigación sobre el entendimiento humano*. Madrid, España: Istmo, 2004.
- [16] R. A. Sarker, H. A. Abbass, and C. S. Newton, *Heuristics and optimization for knowledge discovery*. Australia: Idea Group Pub, 2002.
- [17] L. T. F. Gamut, *Introducción a la lógica*, Eudaba. Buenos Aires, Argentina: Sociedad de Economía Mixta, 2002.
- [18] R. Quillian, "Semantic Memory," in *Semantic Information Processing*, 1968.
- [19] A. Borgida and J. F. Sowa, *Principles of semantic networks: explorations in the representation of knowledge*. San Mateo, California: Morgan Kaufmann, 2014.
- [20] M.-A. Sicilia, *Handbook of Metadata, Semantics and Ontologies*. London: WORLD SCIENTIFIC, 2014.

- [21] L. L. Popejoy *et al.*, “Quantifying care coordination using natural language processing and domain-specific ontology,” *J. Am. Med. Informatics Assoc.*, vol. 59, no. e1, pp. 983–1024, Oct. 2014.
- [22] S. Schulz and L. Jansen, “Formal ontologies in biomedical knowledge representation,” *IMIA Yearb.*, no. 2013: IMIA Yearbook 2013, pp. 132–146, 2013.
- [23] M. Kienle *et al.*, “Concept and development of a unified ontology for generating test and use-case catalogues for assisted and automated vehicle guidance,” *IET Intell. Transp. Syst.*, vol. 8, no. 3, pp. 183–189, May 2014.
- [24] H. Rahimi and H. EL Bakkali, “CIOSOS: Combined Idiomatic-Ontology Based Sentiment Orientation System for Trust Reputation in E-commerce,” in *International Joint Conference*, Á. Herrero, B. Baruque, J. Sedano, Quintián, and H. Corchado, Eds. Burgos, España: Springer, Cham, 2015, pp. 189–200.
- [25] C. Brooksbank, M. T. Bergman, R. Apweiler, E. Birney, and J. Thornton, “The European Bioinformatics Institute’s data resources 2014,” *Nucleic Acids Res.*, vol. 42, no. D1, pp. D18–D25, Jan. 2014.
- [26] B. Y. Xu, H. M. Cai, and C. Xie, “An Ontology Approach for Manufacturing Enterprise Data Warehouses Development,” *Adv. Mater. Res.*, vol. 215, pp. 77–82, Mar. 2011.
- [27] S. Bratus, A. Rumshisky, R. Magar, and P. Thompson, “Using domain knowledge for ontology-guided entity extraction from noisy, unstructured text data,” in *Proceedings of The Third Workshop on Analytics for Noisy Unstructured Text Data - AND '09*, 2009, p. 101.
- [28] S. Staab and R. Studer, *Handbook on Ontologies*. New York, USA: Springer Berlin Heidelberg, 2004.
- [29] R. Neches *et al.*, “Enabling technology for knowledge sharing,” *AI Mag.*, vol.

12, no. 3, p. 36, 1991.

- [30] T. R. Gruber, "A Translation Approach to Portable Ontology Specifications," *Appear. Knowl. Acquis.*, vol. 5, no. 2, pp. 199–220, 1993.
- [31] W. N. Borst, *Construction of Engineering Ontologies for Knowledge Sharing and Reuse*. The Netherlands: Universiteit Twente, 1997.
- [32] R. Studer, V. R. Benjamins, and D. Fensel, "Knowledge engineering: Principles and methods," *Data Knowl. Eng.*, vol. 25, no. 1–2, pp. 161–197, Mar. 1998.
- [33] N. Noy and D. McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology: Knowledge Systems Laboratory, Stanford University," *CiNii*, 2001.
- [34] N. Guarino, "Understanding, building and using ontologies," *Int. J. Hum. Comput. Stud.*, vol. 46, no. 2–3, pp. 293–310, Feb. 1997.
- [35] J. Rumbaugh, I. Jacobson, and G. Booch, *The unified modeling language reference manual*, 2nd ed. Unified Modeling Language Reference Manual, The (2nd Edition): Addison-Wesley, 2005.
- [36] K. Hamilton and R. (Russell) Miles, *Learning UML 2.0*, 1st ed. Beijing, Toronto: O'Reilly, 2006.
- [37] J. B. Warmer and A. G. Kleppe, *The object constraint language : getting your models ready for MDA*. Boston, San Francisco: Addison-Wesley, 2003.
- [38] J. S. Bowman, S. L. Emerson, and M. Darnovsky, *The practical SQL handbook: using structured query language*, 3rd ed. Boston: Addison-Wesley Developers Press, 1996.
- [39] J. R. Campaña Gómez, "Representación y tratamiento semántico de información imprecisa en bases de datos : tesis doctoral," Editorial de la

Universidad de Granada, 2012.

- [40] M.-A. Sicilia and M.-A. Sicilia, *Handbook of metadata, semantics and ontologies*. Spain: World Scientific, 2014.
- [41] N. J. I. (Nicolaas J. I. . Mars and N. International Conference on Building and Sharing Very Large-Scale Knowledge Bases (2nd: 1995: Enschede, *Towards very large knowledge bases : knowledge building & knowledge sharing 1995*. IOS Press, 1995.
- [42] M. Uschold and M. Gruninger, "Ontologies: principles, methods and applications," *Knowl. Eng. Rev.*, vol. 11, no. 2, p. 93, Jun. 1996.
- [43] R. Jasper and M. Uschold, "A Framework for Understanding and Classifying Ontology Applications," in *16th International Joint Conference on Artificial Intelligence (IJCAI'99) Workshop: KRR5: Ontologies and Problem-Solving Methods: Lesson Learned and Future Trends*, 1999, p. 12.
- [44] G. van Heijst, A. T. Schreiber, and B. J. Wielinga, "Using explicit ontologies in KBS development," *Int. J. Hum. Comput. Stud.*, vol. 46, no. 2–3, pp. 183–292, Feb. 1997.
- [45] H.-M. Haav and T.-L. Lubi, "A Survey of Concept-based Information Retrieval Tools on the Web," in *5th East-European Conference, ADBIS 2001*, 2001.
- [46] J. Heflin and J. Hendler, "Dynamic Ontologies on the Web," in *American Association for Artificial Intelligence*, 2000, p. 7.
- [47] J. Kingston, "Multi-perspective ontologies: Resolving common ontology development problems," *Expert Syst. Appl.*, vol. 34, no. 1, pp. 541–550, Jan. 2008.
- [48] R. M. Colomb, "Formal versus Material Ontologies for Information Systems Interoperation in the Semantic Web," *Comput. J.*, vol. 49, no. 1, pp. 4–19, Jul. 2005.

- [49] W. J. Clancey, "The epistemology of a rule-based expert system —a framework for explanation," *Artif. Intell.*, vol. 20, no. 3, pp. 215–251, May 1983.
- [50] W. R. Swartout, "XPLAIN: a system for creating and explaining expert consulting programs," *Artif. Intell.*, vol. 21, no. 3, pp. 285–325, Sep. 1983.
- [51] A. Merino de la Fuente, "Un modelo para la generación de explicaciones en sistemas de recomendación," Universidad Complutense de Madrid, 2012.
- [52] J. Galán, "Metodología para la Generación de Explicaciones para Sistemas de Recomendaciones Sensiboles al Contexto," Centro Nacional de Investigaciones u Desarrollo Tecnológico, 2014.
- [53] P. Sember and I. Zukerman, "Strategies for Generating Micro Explanations for Bayesian Belief Networks," in *Fifth Conference on Uncertainty in Artificial Intelligence*, 2013.
- [54] N. Pennington and R. Hastie, "Explanation-based decision making: Effects of memory structure on judgment.," *J. Exp. Psychol. Learn. Mem. Cogn.*, vol. 14, no. 3, pp. 521–533, 1988.
- [55] A. Tversky and D. Kahneman, "Judgment under Uncertainty: Heuristics and Biases," in *Utility, Probability, and Human Decision Making*, Dordrecht: Springer Netherlands, 1975, pp. 141–162.
- [56] D. B. Rubin, "Causal Inference Using Potential Outcomes: Design, Modeling, Decisions," *J. Am. Stat. Assoc.*, vol. 100, no. 469, pp. 322–331, 2005.
- [57] E. H. Shortliffe, R. Davis, S. G. Axline, B. G. Buchanan, C. C. Green, and S. N. Cohen, "Computer-based consultations in clinical therapeutics: Explanation and rule acquisition capabilities of the MYCIN system," *Comput. Biomed. Res.*, vol. 8, no. 4, pp. 303–320, Aug. 1975.
- [58] F. Costa, S. Ouyang, P. Dolog, and A. Lawlor, "Automatic Generation of

Natural Language Explanations,” Jul. 2017.

- [59] N. Karacapilidis, S. Malefaki, and A. Charissiadis, “A novel framework for augmenting the quality of explanations in recommender systems,” *Intell. Decis. Technol.*, vol. 11, no. 2, pp. 187–197, Jun. 2017.
- [60] W. Chen, D. Aspinall, A. D. Gordon, C. Sutton, and I. Muttik, “A text-mining approach to explain unwanted behaviours,” in *Proceedings of the 9th European Workshop on System Security - EuroSec '16*, 2016, pp. 1–6.
- [61] A. Charissiadis and N. Karacapilidis, “Strengthening the Rationale of Recommendations Through a Hybrid Explanations Building Framework,” in *Intelligent Decision Technologies*, Springer, Cham, 2015, pp. 311–323.
- [62] M. Molina, J. Sanchez-Soriano, and O. Corcho, “Using Open Geographic Data to Generate Natural Language Descriptions for Hydrological Sensor Networks,” *Sensors*, vol. 15, no. 7, pp. 16009–16026, 2015.
- [63] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, “Recommender systems survey,” *Knowledge-Based Syst.*, vol. 46, pp. 109–132, 2013.
- [64] Marco Solimano, “Method for translating a graphical workflow in a textual description,” 2013.
- [65] M. Rodríguez Estrada and M. Márquez Alegría, *Manejo de problemas y toma de decisiones*, 2da ed. Mexico: El Manual Moderno, 2015.
- [66] M. Campos, J. Suárez, R. Ojeda, and R. Ojeda, “Modelo de gestión estratégica para la toma de decisiones en entidades agropecuarias. Implementación en una UBPC del municipio Martí,” *Pastos y Forrajes*, vol. 36, no. 1, pp. 95–105, 2013.
- [67] J. Tejedor, J. López de Heredia, and N. Herranz, “Recomendaciones sobre toma de decisiones y cuidados al final de la vida en neonatología,” *An. Pediatría*, vol. 78, no. 3, p. 190.e1-190.e14, Mar. 2013.

- [68] N. Sáez-Gallego, S. Vila-Maldonado, J. Albellán, and O. Contreras, "Analysis of the visual search behaviour and decision making skill in blocking action in volleyball," *Cuadernos de Psicología del Deporte*, vol. 13, no. 2, Murcia, pp. 31–44, 2013.
- [69] C. Douglas *et al.*, "System and method for presenting content with time based metadata," 2012.
- [70] Leema Mathew, Arun Elias, and Chinnu Rav, "Total Privacy Preservation and Search Quality Improvement in Personalized Web Search - Semantic Scholar," *J. Web Eng.*, vol. 15, no. (5-6), p. 465–483., 2016.
- [71] D. Bačić and A. Fadlalla, "Business information visualization intellectual contributions: An integrative framework of visualization capabilities and dimensions of visual intelligence," *Decis. Support Syst.*, vol. 89, pp. 77–86, Sep. 2016.
- [72] Y. Niwa, S. Shiramatsu, T. Ozono, and T. Shintani, "Developing a Real-Time Web Questionnaire System for Interactive Presentations," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 7, pp. 506–513, 2016.
- [73] S. Costache, V. Viswanathan, H. Aydt, and A. Knollc, "The effect of information uncertainty in road transportation systems," *J. Comput. Sci.*, vol. 16, pp. 170–176, Sep. 2016.
- [74] C. Eichner, T. Nocke, H.-J. Schulz, and H. Schumann, "Interactive Presentation of Geo-Spatial Climate Data in Multi-Display Environments," *ISPRS Int. J. Geo-Information*, vol. 4, no. 2, pp. 493–514, Apr. 2015.
- [75] X. Bai, D. White, and D. Sundaram, "Towards a Flexible User-Centred Visual Presentation Approach," *PACIS 2015 Proc.*, 2015.
- [76] J. C. Beavers *et al.*, "Interactive presentation system," 2014.
- [77] J. Rainer, J. Gómez, and R. Galán, "Generador Automático de

Presentaciones basado en patrones para Robots Guía,” 2013.

- [78] M. Molina and V. Flores, “Generating multimedia presentations that summarize the behavior of dynamic systems using a model-based approach,” *Expert Syst. Appl.*, vol. 39, no. 3, pp. 2759–2770, 2012.
- [79] M. R. Wick and W. B. Thompson, “Reconstructive expert system explanation,” *Artif. Intell.*, vol. 54, no. 1–2, pp. 33–70, Mar. 1992.
- [80] J. Close, “Sistemas de Recomendación,” Universidad Autónoma Metropolitana, 2013.
- [81] T. Gruber and P. Gautier, *Machine-generated explanations of engineering models: A compositional modeling approach*. Citeseer, 1993.
- [82] Y. Iwasaki and H. A. Simon, “Causality in device behavior,” *Artif. Intell.*, vol. 29, no. 1, pp. 3–32, 1986.
- [83] R. Dale and E. Reiter, “Computational Interpretations of the Gricean Maxims in the Generation of Referring Expressions,” *Cogn. Sci.*, vol. 19, no. 2, pp. 233–263, Apr. 1995.
- [84] M. Molina, “Simulating Data Journalism to Communicate Hydrological Information from Sensor Networks,” in *Advances in Artificial Intelligence – IBERAMIA 2012*, 2012, pp. 722–731.
- [85] R. Galli, V. Iker Tolosa, Juan Pedrosa Ginard, and C. G. Antoni, “Escripnauta: sistema de creación y generación de presentaciones multimedia,” Palma de Mallorca, España, 2013.
- [86] M. Bali, *Drools JBoss Rules 5. X Developer’s Guide*. Packt Publishing Ltd, 2013.
- [87] S. Decker *et al.*, “The Semantic Web: the roles of XML and RDF,” *IEEE Internet Comput.*, vol. 4, no. 5, pp. 63–73, 2000.

- [88] P. Browne and P. Johnson, *JBoss Drools business rules*. Olton: Packt Pub, 2009.
- [89] M. Bali, *Drools JBoss rules 5.X developer's guide*, 1st ed. Livery: Packt Pub, 2013.
- [90] A. J. Bouwer, *Explaining behaviour: using qualitative simulation in interactive learning environments*. SIKS, 2005.
- [91] W. C. Mann and S. A. Thompson, "Rhetorical structure theory: Toward a functional theory of text organization," *Text-Interdisciplinary J. Study Discourse*, vol. 8, no. 3, pp. 243–281, 1988.
- [92] M. Taboada and W. C. Mann, "Applications of Rhetorical Structure Theory," *Discourse Stud.*, vol. 8, no. 4, pp. 567–588, Aug. 2006.
- [93] C. Demergasso *et al.*, "From Knowledge to Best Practices in Bioleaching," *Solid State Phenom.*, vol. 262, pp. 285–289, Aug. 2017.
- [94] COCHILCO, "Biolixiviación: Desarrollo Actual y sus Expectativas," COCHILCO, Comisión Chilena del Cobre, Chile, 2009.
- [95] P. Soto, P. Galleguillos, M. Serón, V. Zepeda, C. Demergasso, and C. Pinilla, "Parameters influencing the microbial oxidation activity in the industrial bioleaching heap at Escondida mine, Chile," *Hydrometallurgy*, vol. 133, pp. 51–57, 2013.
- [96] R. Fredes, "Biolixiviación a Baja Temperatura - Evaluación de Dos Alternativas de Inoculación con un Consorcio Bacteriano Psicrotolerante," 2012.
- [97] T. Walter, F. S. Parreiras, and S. Staab, "An ontology-based framework for domain-specific modeling," *Softw. Syst. Model.*, vol. 13, no. 1, pp. 83–108, Feb. 2014.

- [98] J. M. Rodríguez, "Métodos de Investigación Cualitativa," *Rev. Investig. Silogismo*, vol. 1, no. 8, 2011.
- [99] J. L. Mate Hernández and J. Pazos Sierra, *Ingeniería del conocimiento: diseño y construcción de sistemas expertos*. Sociedad para Estudios Pedagógicos Argentinos, 1988.
- [100] C. Robson and K. McCartan, *Real world research: a resource for users of social research methods in applied settings*. 2016.
- [101] F. Tassario and T. Alexandre, "Trapezio – Translation Post Editor: um ambiente de pós- edición de traducciones automáticas."
- [102] V. Flores, "Generación Automática de Resúmenes de Comportamiento de Sistemas Dinámicos Mediante Modelos de Conocimiento del Dominio.," Universidad Politécnica de Madrid, Madrid, 2010.
- [103] R. Turner, S. Sripada, and E. Reiter, "Generating Approximate Geographic Descriptions," Springer Berlin Heidelberg, 2010, pp. 121–140.
- [104] M. Genero Bocco, J. A. Cruz Lemus, and M. G. Piattini Velthuis, *Métodos de investigación en ingeniería del software*. Ra-Ma, 2014.
- [105] GamerDic, "Renderizar | Definición en GamerDic | Diccionario de términos sobre videojuegos y cultura gamer," 2017. [Online]. Available: <http://www.gamerdic.es/termino/renderizar>. [Accessed: 23-Sep-2017].

# ANEXOS

---

## Anexo I

### Encuesta para aplicar técnica Post-Edición

Estimado, a continuación, se les presentan los datos con los que el Sistema Experto llega a su conclusión, además de la explicación que genera SAGE (sistema automático de generación de explicaciones). Según esta información, se le solicita indicar en cada una de las explicaciones que palabras omitiría y cuales adicionaría. Se le agradece mantener en lo posible el esquema general del texto original.

#### **Franja 110 081007 (Regla 1)**

Cantidad de Microorganismos: 4389

Temperatura de la Muestra: 20,5 °C

% de termófilos moderados: 82,7%

#### **La temperatura interna no pudo ser estimada**

La población de microorganismos no supera la cantidad de 10e5, para este caso el valor es de 4389.

#### **Franja 403 150130 (Regla 2)**

Cantidad de Microorganismos: 3057195

Temperatura de la Muestra: 29,8 °C

% de termófilos moderados: 99,9%

#### **Las condiciones de operación utilizadas para la Biolixiviación en la franja determinan una temperatura interna de 52°C**

La temperatura interna obtenida se debe a que la suma de los microorganismos es mayor a 10e5, para este caso es de 3057195. Además, la temperatura del PLS supera los 20 °C, alcanzando el valor de 29,8. La temperatura interna obtenida se debe además de los factores descrito anteriormente a que en la franja habitan más de 52% de microorganismos clasificados como termófilos moderados. Específicamente la franja cuenta con una población de termófilos moderados del 99,9%

#### **Franja 119 100630 (Regla 3)**

Cantidad de Microorganismos: 1292270

Temperatura de la Muestra: 25,4 °C

% de termófilos moderados: 22,5%

#### **La temperatura interna estimada es de 35°C**

La temperatura interna pudo ser estimada ya que la cantidad de microorganismos supera 10e5, específicamente se estima una población de 1292270. Además, la temperatura del PLS supera los 20 °C, alcanzando el valor de 25,4 °C. El porcentaje de microorganismos clasificados como termófilos moderados representa menos del 52% del total, en concreto para la franja analizada hay 22,5%.

#### **Franja 313 130821 (Regla 4)**

Cantidad de Microorganismos: 2863410

Temperatura de la Muestra: 0 °C

% de termófilos moderados: 99,5%

**La temperatura interna estimada es de 0°C**

La temperatura interna obtenida se debe a que la suma de los microorganismos es mayor a  $10^5$ , para este caso es de 2863410. Además de que la temperatura del no superó los 20°C, solamente alcanzó el valor de 0 °C.

**¡MUCHAS GRACIAS!!!!**

## Anexo II

### Encuesta aplicada a los expertos

Estimado, a continuación, se le presentan cuatro de las explicaciones obtenidas automáticamente por el sistema automático de generación de explicaciones (SAGE). Según esta información, se le solicita calificar estas explicaciones utilizando una escala del 1 al 5 siendo el número 1 el extremo negativo y el número 5 el positivo. Los parámetros a evaluar son:

- **Cobertura:** esta métrica es la encargada de medir el grado de soporte (cobertura) ofrecida por la explicación. Se le pedirá al experto que plasme su opinión sobre la presencia de todas las variables que deben estar según los datos que maneja el Sistema Experto.
- **Claridad:** esta métrica es la encargada de recoger la claridad, nitidez y fluidez de la explicación.
- **Coherencia:** esta métrica registrará el grado de coherencia de la explicación con los fenómenos representados por el Sistema Experto.

Las explicaciones a evaluar son:

#### La temperatura interna no pudo ser estimada

La población de microorganismos no supera la cantidad de  $10e5$ , para este caso el valor es de 4389.

Explicación 1:

Cobertura	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>
Claridad	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>
coherencia	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>

#### Las condiciones de operación utilizadas para la Biolixiviación en la franja determinan una temperatura interna de 52°C

La temperatura interna obtenida se debe a que la suma de los microorganismos es mayor a  $10e5$ , para este caso es de 3057195. Además, la temperatura del PLS supera los 20 °C, alcanzando el valor de 29,8. La temperatura interna obtenida se debe además de los factores descrito anteriormente a que en la franja habitan más de 52% de microorganismos clasificados como termófilos moderados. Específicamente la franja cuenta con una población de termófilos moderados del 99,9%

Explicación 2:

Cobertura	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>
Claridad	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>
coherencia	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>

#### La temperatura interna estimada es de 35°C

La temperatura interna pudo ser estimada ya que la cantidad de microorganismos supera  $10e5$ , específicamente se estima una población de 1292270. Además, la temperatura del PLS supera los 20 °C, alcanzando el valor de 25,4 °C. El porcentaje de microorganismos clasificados como termófilos moderados representa menos del 52% del total, en concreto para la franja analizada hay 22,5.

Explicación 3:

Cobertura	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>
Claridad	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>
coherencia	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>

**La temperatura interna estimada es de 0°C**

La temperatura interna obtenida se debe a que la suma de los microorganismos es mayor a  $10^5$ , para este caso es de 2863410. Además de que la temperatura del PLS no superó los 20°C, solamente alcanzó el valor de 0 °C.

Explicación 4:

Cobertura	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>
Claridad	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>
coherencia	1 <input type="radio"/>	2 <input type="radio"/>	3 <input type="radio"/>	4 <input type="radio"/>	5 <input type="radio"/>

**¡MUCHAS GRACIAS!!!!!!**

### Anexo III

#### Encuesta aplicada a potenciales usuarios del SAGE

Estimado, a continuación, se le presentan un conjunto de preguntas relacionadas con el sistema automático de generación de explicaciones (SAGE), seleccione la respuesta que ud encuentre adecuada

¿La interfaz del prototipo para la generación de explicaciones automáticas es fácil de usar?

Si	<input type="radio"/>
Más bien si	<input type="radio"/>
De dificultad media	<input type="radio"/>
Más bien no	<input type="radio"/>
Absolutamente no	<input type="radio"/>

La documentación (documento de tesis) que acompaña al prototipo es

Muy útil	<input type="radio"/>
Más bien útil	<input type="radio"/>
Normal	<input type="radio"/>
Más bien inútil	<input type="radio"/>
Totalmente inútil	<input type="radio"/>

La explicación que muestra el prototipo es

Exactamente lo que esperaba	<input type="radio"/>
Buena, pero le agregaría detalles	<input type="radio"/>
Dista de lo que esperaba	<input type="radio"/>
Tota distinta de lo que esperaba	<input type="radio"/>

¿La forma de insertar las plantillas de textos es fácil de usar?

Si	<input type="radio"/>
Más bien si	<input type="radio"/>
De dificultad media	<input type="radio"/>
Más bien no	<input type="radio"/>
Absolutamente no	<input type="radio"/>

La explicación brindada por el prototipo de generación automática de explicaciones

Parece ser escrita por una persona	<input type="radio"/>
Se nota que no es escrita por un humano, pero se asemeja bastante	<input type="radio"/>
Es estática y se nota que son plantillas de textos fijas	<input type="radio"/>

**¡MUCHAS GRACIAS!!!!!!**

## Anexo IV: RDF/XML con el modelado del dominio del Biolixiviación.

```
<?xml version="1.0"?>
<rdf:RDF xmlns="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#"
  xml:base="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <owl:Ontology rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17"/>

  <!--
  //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
  //
  // Object Properties
  //
  //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
  -->

  <!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TieneUn -->

  <owl:ObjectProperty rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TieneUn">
    <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Microorganismo"/>
    <rdfs:range rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Dominio"/>
    <rdfs:range rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Especie"/>
    <rdfs:range rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Genero"/>
  </owl:ObjectProperty>

  <!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMuestra -->

  <owl:ObjectProperty rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMuestra">
    <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Muestra"/>
    <rdfs:range rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#ElectrolitoPobre"/>
    <rdfs:range rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#NoDefinido"/>
    <rdfs:range rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLS"/>
    <rdfs:range rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLScomun"/>
    <rdfs:range rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Refino"/>
  </owl:ObjectProperty>

  <!--
  //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
  //
  // Data properties
  //
  //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
  -->

  <!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Cantidad -->
```

```

<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Cantidad">
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#A_Thiooxidans"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Acidithiobacillus"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Arquea"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Bacteria"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Eukarya"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum_SP"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Mesofilo"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#No_Definido"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilo"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilos_Moderados"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termotolerante"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Todos"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#long"/>
  <rdfs:comment>La suma de</rdfs:comment>
</owl:DatatypeProperty>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Edad -->

<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Edad">
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#ElectrolitoPobre"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#NoDefinido"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLS"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLScomun"/>
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Refino"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#integer"/>
</owl:DatatypeProperty>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Estado -->

<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Estado">
  <rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#EstimacionTemperatura"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
</owl:DatatypeProperty>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Porcentaje -->

<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Porcentaje">

```

```

<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#A_Thiooxidans"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Acidithiobacillus"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Arquea"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Bacteria"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Eukarya"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum_SP"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Mesofilo"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#No_Definido"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilo"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilos_Moderados"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termotolerante"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Todos"/>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
<rdfs:comment>El porcentaje de</rdfs:comment>
</owl:DatatypeProperty>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#RangoTemperatura -->

<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#RangoTemperatura">
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#EstimacionTemperatura"/>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Temperatura -->

<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Temperatura">
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#ElectrolitoPobre"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#NoDefinido"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLS"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLScomun"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Refino"/>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#integer"/>
</owl:DatatypeProperty>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Tolerancia -->

<owl:DatatypeProperty rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Tolerancia">
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#A_Thiooxidans"/>

```

```

<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Acidithiobacillus"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Arquea"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Bacteria"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Eukarya"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum_SP"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Mesofilo"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#No_Definido"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilo"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilos_Moderados"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termotolerante"/>
<rdfs:domain rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Todos"/>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

<!--
////////////////////////////////////
//
// Classes
//
////////////////////////////////////
-->

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#A_Thiooxidans -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#A_Thiooxidans">
<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Especie"/>
<owl:disjointWith rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum_SP"/>
<rdfs:comment>valor 1</rdfs:comment>
<rdfs:comment>valor 2</rdfs:comment>
<rdfs:comment>valor 3</rdfs:comment>
<rdfs:comment>valor conflictivo</rdfs:comment>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Acidithiobacillus -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Acidithiobacillus">
<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Genero"/>
<owl:disjointWith rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Arquea -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-

```

```

17#Arquea">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Dominio"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Bacteria -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Bacteria">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Dominio"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Dominio -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Dominio">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Microorganismo"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#ElectrolitoPobre -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#ElectrolitoPobre">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMuestra"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Especie -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Especie">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Microorganismo"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#EstimacionTemperatura -->
>

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#EstimacionTemperatura">
  <rdfs:comment>La temperatura interna estimada es de</rdfs:comment>
  <rdfs:comment>La temperatura interna estimada para la franja es de</rdfs:comment>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string">La temperatura interna no pudo ser estimada</rdfs:comment>
  <rdfs:comment>Las condiciones de operacion utilizadas para la Biolixiviacion en la franja determinan una temperatura interna de</rdfs:comment>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string">No se pudo estimar la temperatura interna</rdfs:comment>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Eukarya -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Eukarya">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Dominio"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Genero -->

```

```

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Genero">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Microorganismo"/>
  <rdfs:comment>ppppp</rdfs:comment>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Genero"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum_SP -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Leptospirillum_SP">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Especie"/>
  <rdfs:comment>lolo</rdfs:comment>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Mesofilo -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Mesofilo">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMicroorganismo"/>
  <rdfs:comment>yhyhy</rdfs:comment>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Microorganismo -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Microorganismo">
  <owl:equivalentClass rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMicroorganismo"/>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#long">El porciento de microorganismos clasificados como termofilos moderados representa mas del 52 % del total, en concreto para la franja analizada hay</rdfs:comment>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#int">El porciento de microorganismos clasificados como termofilos moderados representa menos del 52 % del total, en concreto para la franja analizada hay</rdfs:comment>
  <rdfs:comment rdf:datatype="http://www.w3.org/2000/01/rdf-schema#Literal">La cantidad de microorganismos no supera 10e5, para este caso el valor es de</rdfs:comment>
  <rdfs:comment rdf:datatype="http://www.w3.org/2000/01/rdf-schema#Literal">La poblacion de micoorganismos no supera la cantidad de 10e5, para este caso el valor es de</rdfs:comment>
  <rdfs:comment>La temperatura intena pudo ser estimada ya que la cantidad de miroorganismos supera 10e5, especificamente se estima una poblacion de</rdfs:comment>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#int">La temperatura interna estimada se obtuvo guiandonos ademas de los factores anteriores a la poblacion de microorganismos termofilos moderados que es de</rdfs:comment>
  <rdfs:comment>La temperatura interna obtenida se debe a que la suma de los microorganismmos es mayor a 10e5, para este caso es de</rdfs:comment>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#long">La temperatura interna obtenida se debe ademas de los factores descrito anteriormente a que en la franja habitan mas del 52% de microorganismos clasificados como termofilos moderados. Especificamente la franja cuenta con una poblacion de termofilos moderados de</rdfs:comment>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#int">La temperatura interna obtenida se debe ademas de los factores descrito anteriormente a que en la franja habitan menos del 52% de

```

microorganismos clasificados como termofilos moderados. Especificamente la franja cuenta con una poblacion de termofilos moderados de</rdfs:comment>

</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Muestra -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Muestra">

<owl:equivalentClass rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMuestra"/>

<rdfs:comment>Ademas la temperatura del PLS no supero los 20 grados celsius, solamente alcanzo el valor de</rdfs:comment>

<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#long">Ademas la temperatura del PLS supera los 20 grados celsius, alcanzando el valor de</rdfs:comment>

<rdfs:comment>Se puede decir que la temperatura interna estimada alcanzo el valor declarado ya que la temperatura del PLS no supero los 20 grados celsius. En esta medicion la temperatura del PLS alcanzo</rdfs:comment>

<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#long">Se puede decir que la temperatura interna estimada alcanzo el valor declarado ya que la temperatura del PLS supero los 20 grados celsius. En esta medicion la temperatura del PLS alcanzo</rdfs:comment>

</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#NoDefinido -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#NoDefinido">

<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMuestra"/>

</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#No\_Definido -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#No\_Definido">

<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMicroorganismo"/>

</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLS -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLS">

<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMuestra"/>

<rdfs:comment></rdfs:comment>

</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLScomun -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLScomun">

<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMuestra"/>

</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Refino -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Refino">

<rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMuestra"/>

</owl:Class>

```
<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilo -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilo">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMicroorganismo"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilos_Moderados -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilos_Moderados">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMicroorganismo"/>
  <rdfs:comment>El por ciento de microorganismos de tipo Termófilos Moderados recolectados es de</rdfs:comment>
  <rdfs:comment>La población de microorganismos de tipo Termófilos Moderados es menor al 52% del total de microorganismos recolectados en la muestra</rdfs:comment>
  <rdfs:comment>Los etermófilo</rdfs:comment>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termotolerante -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termotolerante">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMicroorganismo"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMicroorganismo -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMicroorganismo">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Microorganismo"/>
  <rdfs:comment>Los tipos de</rdfs:comment>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMuestra -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMuestra">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Muestra"/>
</owl:Class>

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Todos -->

<owl:Class rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Todos">
  <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#TipoMicroorganismo"/>
  <rdfs:comment>Todos</rdfs:comment>
</owl:Class>

<!--
////////////////////////////////////
//
// Individuals
//
////////////////////////////////////
-->
```

```

<!-- http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#A1 -->

<owl:NamedIndividual rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#A1">
  <rdf:type rdf:resource="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilos_Moderados"/>
</owl:NamedIndividual>

<!--
////////////////////////////////////
//
// General axioms
//
////////////////////////////////////
-->

<rdf:Description>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AllDisjointClasses"/>
  <owl:members rdf:parseType="Collection">
    <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Arquea"/>
      <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Bacteria"/>
        <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Eukarya"/>
          </owl:members>
    </rdf:Description>
  </rdf:Description>
  <rdf:Description>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AllDisjointClasses"/>
    <owl:members rdf:parseType="Collection">
      <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#ElectrolitoPobre"/>
        <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#NoDefinido"/>
          <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLS"/>
            <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#PLScomun"/>
              <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Refino"/>
                </owl:members>
            </rdf:Description>
          </rdf:Description>
        <rdf:Description rdf:about="http://www.w3.org/2002/07/owl#AllDisjointClasses"/>
        <owl:members rdf:parseType="Collection">
          <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Mesofilo"/>
            <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilo"/>
              <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termofilos_Moderados"/>
                <rdf:Description rdf:about="http://www.semanticweb.org/yahima/ontologies/2016/10/untitled-ontology-17#Termotolerante"/>
                  </owl:members>
                </rdf:Description>
              </rdf:Description>
            </rdf:RDF>
          </owl:members>
        </rdf:Description>
      </owl:members>
    </rdf:Description>
  </rdf:RDF>

<!-- Generated by the OWL API (version 4.2.5.20160517-0735) https://github.com/owlcs/owlapi -->

```

## Anexo V: Fichero FileLogger para la celda 110

```
<object-stream>
<org.drools.core.audit.WorkingMemoryLog>
<version>6.1</version>
<events>
  <org.drools.core.audit.event.ObjectLogEvent>
    <type>1</type>
    <factId>289</factId>
    <objectToString>Microorganismo.cantidad[value=4389]</objectToString>
  </org.drools.core.audit.event.ObjectLogEvent>
  <org.drools.core.audit.event.ObjectLogEvent>
    <type>1</type>
    <factId>290</factId>
    <objectToString>Muestra.temperatura[value=20,5]</objectToString>
  </org.drools.core.audit.event.ObjectLogEvent>
  <org.drools.core.audit.event.ObjectLogEvent>
    <type>1</type>
    <factId>291</factId>
    <objectToString>Microorganismo.porcentaje[value=82,7]</objectToString>
  </org.drools.core.audit.event.ObjectLogEvent>
  <org.drools.core.audit.event.ObjectLogEvent>
    <type>1</type>
    <factId>292</factId>
    <objectToString>EstimacionTemperatura.estado[value=false]</objectToString>
  </org.drools.core.audit.event.ObjectLogEvent>
  <org.drools.core.audit.event.ActivationLogEvent>
    <type>4</type>
    <activationId>Si la suma de microorganismos es menor que 10e5 [289, 292]</activationId>
    <rule>Si la suma de microorganismos es menor que 10e5</rule>
    <declarations>$tempEstimada=false</declarations>
    <factHandleIds>292,291</factHandleIds>
  </org.drools.core.audit.event.ActivationLogEvent>
  <org.drools.core.audit.event.ActivationLogEvent>
    <type>6</type>
    <activationId>Si la suma de microorganismos es menor que 10e5 [289, 292]</activationId>
    <rule>Si la suma de microorganismos es menor que 10e5</rule>
    <declarations>$tempEstimada=false</declarations>
    <factHandleIds>292,291</factHandleIds>
  </org.drools.core.audit.event.ActivationLogEvent>
  <org.drools.core.audit.event.ObjectLogEvent>
    <type>2</type>
    <factId>292</factId>
    <objectToString>EstimacionTemperatura.estado["No se puede estimar"]</objectToString>
  </org.drools.core.audit.event.ObjectLogEvent>
  <org.drools.core.audit.event.ActivationLogEvent>
    <type>7</type>
    <activationId>Si la suma de microorganismos es menor que 10e5 [289, 292]</activationId>
    <rule>Si la suma de microorganismos es menor que 10e5</rule>
    <declarations>$tempEstimada="No se puede estimar"/</declarations>
    <factHandleIds>292,291</factHandleIds>
  </org.drools.core.audit.event.ActivationLogEvent>
  <org.drools.core.audit.event.ObjectLogEvent>
    <type>3</type>
    <factId>291</factId>
    <objectToString>Microorganismo.porcentaje[value=0]</objectToString>
  </org.drools.core.audit.event.ObjectLogEvent>
  <org.drools.core.audit.event.ObjectLogEvent>
    <type>3</type>
    <factId>290</factId>
    <objectToString>Muestra.temperatura[value=0]</objectToString>
  </org.drools.core.audit.event.ObjectLogEvent>
```

```
<org.drools.core.audit.event.ObjectLogEvent>
  <type>3</type>
  <factId>289</factId>
  <objectToString>Microorganismo.cantidad[value=0]</objectToString>
</org.drools.core.audit.event.ObjectLogEvent>
<org.drools.core.audit.event.ObjectLogEvent>
  <type>3</type>
  <factId>292</factId>
  <objectToString>EstimacionTemperatura.estado[value=false]</objectToString>
</org.drools.core.audit.event.ObjectLogEvent>
</events>
<engine>PHREAK</engine>
</org.drools.core.audit.WorkingMemoryLog>
</object-stream>
```