



**Universidad  
de La Laguna**

**TRABAJO FIN DE GRADO**

**Grado En Ingeniería Electrónica**

**Industrial y Automática**

---

# **Estructura Ontológica de una Instalación Eléctrica**

---

***Nombre: Paul Arinzechukwu Eloka***

***Tutor: Evelio José González González***

# ÍNDICE

<b>1. INTRODUCCIÓN</b>	<b>4</b>
<b>2. DESCRIPCIÓN Y OBJETIVOS DEL PROYECTO</b>	<b>5</b>
<b>3. WEB SEMÁNTICA</b>	<b>5</b>
3.1. INTRODUCCIÓN	5
3.2. DESARROLLO	6
3.2.1 ¿QUÉ ES LA WEB SEMÁNTICA?	7
3.3. METADATOS	8
3.4. ONTOLOGÍA	8
3.4.1. DEFINICIÓN DE ONTOLOGÍA	8
3.4.2. CARACTERÍSTICAS DE UNA ONTOLOGÍA	9
3.5. TECNOLOGÍA UTILIZADA EN LA WEB SEMÁNTICA	11
<b>4. PROGRAMA Y LENGUAJE UTILIZADOS PARA LA ONTOLOGÍA</b>	<b>12</b>
4.1. PROTÉGÉ	12
4.2. LENGUAJE OWL	13
<b>5. LA INSTALACIÓN ELÉCTRICA Y SUS COMPONENTES SEGÚN LA REBT</b>	<b>14</b>
5.1. COMPONENTES DE UNA INSTALACIÓN ELÉCTRICA	14
5.2. INSTALACIÓN ELÉCTRICA REALIZADO EN PROTÉGÉ PARA EL PROYECTO	16
5.2.1. COMPONENTES DE LA INSTALACIÓN ELÉCTRICA	16
<b>6. COMPONENTES DE ONTOLOGÍA OWL UTILIZADAS EN EL PROGRAMA PARA LA CONSTRUCCIÓN DE LA ONTOLOGÍA</b>	<b>18</b>
6.1. CLASES	19
6.2. PROPIEDADES DE OWL	21
6.2.1. CARACTERÍSTICAS DE LAS PROPIEDADES	22
6.2.2. PROPIEDADES DE OBJETOS	24
6.2.3. PROPIEDADES DE DATOS	29
6.3. INDIVIDUOS	32
6.4. RAZONADORES EN PROTÉGÉ	35
6.5. LENGUAJE DE REGLAS DE LA WEB SEMÁNTICA (SWRL)	36
6.5.1. SWRL EN PROTÉGÉ SIN REGLAS	37
6.5.2. SWRL EN PROTÉGÉ CON REGLAS	41
<b>7. PROBLEMAS A LO LARGO DE LA EJECUCIÓN DEL PROGRAMA Y SUS SOLUCIONES</b>	<b>46</b>
<b>8. PRESUPUESTO DEL PROYECTO</b>	<b>48</b>
<b>9. CONCLUSIONES</b>	<b>49</b>
<b>REFERENCIAS</b>	<b>50</b>

# ABSTRACT

Currently, the world is in a digitized time, where almost everything is done virtually and therefore there is an overwhelming amount of data on the web that makes it difficult to access specific information. This made the scientist Tim Berners-Lee to consider how to improve the cooperation between man and computer and to allow despite the amount of information that there is, the capability to find the appropriate answers to our search. This is how the idea of the Semantic Web arose, whose purpose is to provide better defined and linked data on the web, which allows the computer to interpret the information without human need. Knowledge entered on the web must be legible for computers, and that's why the Semantic Web uses ontologies for the representation of the knowledge. This ontology will help the computer to be able to interpret and reason the searches about an electrical installation with greater precision.

Therefore, as the name of the project itself indicates, the ontology of an electrical installation will be made to give computers the ability to structure and handle information based on the semantic assessment of its contents, so that the artificial intelligence can understand the meaning of the content.

Thus, the "Protégé" program will be used for this ontology through the OWL language.

Keywords: Ontology, Semantic Web, Electrical Installation, Protégé, OWL Language.

# RESUMEN

Actualmente, el mundo se encuentra en una era digitalizada, donde casi todo se realiza de forma virtual y por ello hay una abrumadora cantidad de datos en la web que dificulta el acceso a informaciones específicas. Esto hizo que el científico Tim Breneers-Lee se plantee la manera de mejorar la cooperación entre el hombre y el ordenador y así permitir que a pesar de la cantidad de información que hay, este sea capaz de encontrar las respuestas adecuadas a nuestra búsqueda. Así surgió la idea de la Web Semántica cuya finalidad es la de proporcionar en la web datos mejor definidos y enlazados, que permita al ordenador interpretar las informaciones sin necesidad humana. El conocimiento introducido en la web debe ser legible para los ordenadores, por lo que la Web Semántica hace uso de las ontologías para la representación del conocimiento.

Por ello, como el propio nombre del proyecto indica, se va a hacer la ontología de una instalación eléctrica para dar a los ordenadores la capacidad de estructurar y manejar la información en base a la valoración semántica de sus contenidos, con el fin de que la inteligencia artificial pueda entender el significado del contenido. Dicha ontología, servirá para que el ordenador sea capaz de interpretar y razonar las búsquedas acerca de una instalación eléctrica con mayor precisión

Así se hará uso del programa de “Protégé” para dicha ontología a través del lenguaje OWL.

Palabras Claves: Ontología, Web Semántica, Instalación Eléctrica, Protégé, Lenguaje OWL.

# 1. INTRODUCCIÓN

El presente proyecto se ha redactado como Trabajo Fin De Grado en para obtener la titulación en el grado de Ingeniería Electrónica Industrial y Automática.

La ontología de una instalación eléctrica es un estudio necesario para los grados de ingeniería y otros grados relacionados con la tecnología puesto que estamos rodeados de instalaciones eléctricas y es necesario saber un mínimo acerca de ellos. El motivo de este estudio es el de concienciar al lector acerca del funcionamiento y los elementos que componen una instalación eléctrica, y también que a la hora de buscar cualquier información más específica acerca de una este, le sea mucho más fácil a pesar de la abrumadora cantidad de información que hay acerca de ella.

Este trabajo se basa como su propio título indica, en el estudio de la ontología (abstracción más o menos esquemática) de una instalación eléctrica, que según la Normativa de instalaciones eléctricas se define como el conjunto de componentes entre una fuente de alimentación y los consumidores.

¿Por qué una ontología? Pues también se verá los pasos que han llevado elegir este sustantivo. Pero todo se remonta a la web semántica propuesta por el científico de computación Tim Berners-Lee.

## 2. DESCRIPCIÓN Y OBJETIVOS DEL PROYECTO

Este proyecto se trata de la realización de la ontología de una instalación eléctrica, en la que se utilizara el programa del Protégé para realizar dicha ontología. Se entenderá el significado de una ontología y su relación con la web semántica. Que viene a ser que para que la web semántica pueda cumplir sus objetivos necesita de las ontologías ya que aporta los recursos necesarios para definir cualquier dominio, tanto los conceptos de estos como las relaciones existentes y la jerarquía que los sustenta. Luego se hará uso del Protégé, implementando los conceptos, relaciones, reglas... necesarias para la construcción de una instalación eléctrica. Este dominio se ha seleccionado puesto que es un concepto muy importante en el ámbito de la ingeniería electrónica industrial, al ser necesario para suministrar energía eléctrica a todo el mundo (viviendas, industrias, locales...)

Los objetivos de proyecto son:

- Entender que es una ontología y por qué se necesita para este proyecto
- Familiarizarse con el programa Protégé
- Conocer y entender el concepto de la web semántica
- Relación entre ontología y web semántica
- Aprender todo lo necesario acerca de una instalación eléctrica
- Realizar una ontología con el programa Protégé, que sirva para cualquier instalación eléctrica
- Aprender el lenguaje de reglas de la web semántica (SWRL) y cómo aplicarlos.

## 3. WEB SEMÁNTICA

Antes de adentrarnos en el tema principal del proyecto, hay que saber de donde surge la web semántica, entender qué es y cómo funciona, ya que de ella deriva este proyecto.

### 3.1. INTRODUCCIÓN

El creador de la World Wide Web (WWW), Tim Berners-Lee, propuso una solución basada en un sistema distribuido de hipertextos con la idea de solventar los problemas de la pérdida de la información, permitiendo así el intercambio de información entre los científicos del centro de investigación de la CERN, ya que le preocupaba la gestión de la información acerca de los aceleradores y experimentos que se hacían en dicho centro.

Así surgió el identificador Uniforme de Recurso (URI), usado para identificar un recurso sin equivocación, ya sea una página, un servicio o una dirección de correo en una red o sistema. Y para la Transferencia de Hipertextos (HTTP), también surgió el Protocolo para el intercambio de hipertextos y el Lenguaje de Marcado de Hipertextos (HTML) para la presentación de la información.

En 1990 Tim Berners-Lee acabó con el primer servidor web siendo el primer usuario y un año después publicó el primer borrador de las especificaciones de HTML y HTTP.

La Web 1.0, que fue la primera versión de la web, duró desde 1990 hasta 2000, y consistía en páginas HTML editadas manualmente y que se enlazaban con otras páginas, que a menudo eran escritas por personas con los mismos intereses.

## **3.2. DESARROLLO**

Los tiempos en los que vivimos actualmente ha convertido a la sociedad en una era digital, en la que el internet está presente en todos los ámbitos de la vida. Aplicaciones como la web, las redes sociales, el correo electrónico, servicios de la banca online, o incluso programas de presentación, se han convertido en herramientas indispensables para el ser humano. Por ello el uso de la web forma parte del día a día en la sociedad actual sobre todo para el intercambio de información, convirtiéndose así en uno de los medios de publicación más importante de la sociedad. Por ello la web actual se ha desarrollado más que la idea inicial de Tim Berners-Lee, que era como se ha comentado anteriormente para el intercambio de información entre los científicos. A dicha web se le ha agregado nuevas herramientas gracias al desarrollo de lenguajes de programación y la integración de sistemas de bases de datos. Ahora solo basta con un navegador web para que el usuario pueda crear todo tipo de herramientas para uso propio o también con la opción de compartirla con todo el mundo. Esta nueva generación de herramientas se conoce como la Web 2.0 o web social en la que los usuarios comienzan a interactuar entre sí creando y compartiendo contenidos textuales y audiovisuales con facilidad.

Esto ha hecho que internet se convierta en un almacén desbordado de información, en la que hay tal cantidad de recursos desorganizados, duplicados o desactualizados que a veces resulta difícil encontrar ciertas informaciones de nuestro interés. El problema es que aún estamos contribuyendo a dicho crecimiento y aunque los buscadores de la web han mejorado con los años, aún nos alejamos más de esa búsqueda más exacta de informaciones específicas.

Esto se debe a que la calidad, estructuración y originalidad de contenidos en la web no se han desarrollado conforme lo han ido haciendo los procesos de publicación de estos. Hay una gran cantidad de duplicados porque los usuarios han preferido copiar contenidos de otros en lugar de referenciarlos con enlaces de hipertexto. El

mal uso de metadatos HTML en las páginas, hacen que se distorsione su utilidad en los procesos de búsqueda y que tampoco sea posible distinguir el tipo de recurso recuperado durante la búsqueda.

Se precisa del uso de metadatos para la recuperación de información en la web, aplicando modelos estándares para la descripción de los recursos. Aparte, su desarrollo y uso ayudarían en la mejora de los buscadores web y aumentaría la visión de la web para el intercambio y procesamiento de datos de manera automática.

### 3.2.1 ¿QUÉ ES LA WEB SEMÁNTICA?

La web semántica no se difiere a la idea original desarrollada por Tim Berners-Lee, y al igual que la web 2.0 se rige por el uso de un conjunto de herramientas y tecnologías. La web semántica se ideó con el propósito de superar las carencias de la web actual, y para ello propone una estructuración mejor definida de los contenidos y servicios de la web. De esta manera lo que busca la web semántica es clasificar, dotar de estructura y anotar los recursos con semántica explícita procesable por máquinas, ya que la web actual carece de una organización clara, posee una semántica implícita y un crecimiento caótico de recursos que va aumentando.

La web semántica mantiene los principios de la web desarrollado por Tim Berners-Lee, como los principios de descentralización, compatibilidad, compartición, facilidad de acceso y contribución. Pero el problema es conseguir que usuarios, desarrolladores y programa se entiendan entre sí, para así facilitar la comunicación entre ellos. Y para conseguir abordar este problema, la web semántica recurre a la construcción de ontologías del campo de inteligencia artificial.

Con esto la definición de la web semántica nos la ofrecen Hendler, Berners-Lee y Miller (2002):

*“La web semántica es una extensión de la actual web en la que a la información disponible se le otorga un significado bien definido que permita a los ordenadores y las personas trabajar en cooperación. Está basada en la idea de proporcionar en la web datos definidos y enlazados, permitiendo que aplicaciones heterogéneas localicen, integren, razonen y reutilicen la información presente en la web.”* (Tim Berners-Lee, citado por Juan Antonio Pastor Sanchez, 2011, pag 15-16)

Similar definición es la que ofrece Berners-Lee junto con Miller (2002), en la que también exponen el modo en el que el W3C1 coordina la consecución de estos objetivos:



*“La web semántica es una extensión de la actual web en la que a la información disponible se le otorga un significado bien definido que permita a los ordenadores y las personas trabajar en cooperación. La W3C Semantic Web Activity, en colaboración con investigadores y socios industriales, se encarga de la definición de estándares y tecnologías que permitan a los datos de la web ser definidos y enlazados de forma que puedan ser usados para una localización más eficaz, automatización, integración y reutilización entre aplicaciones.” (Tim Berners-Lee, citado por Juan Antonio Pastor Sanchez, 2011, pag 15-16)*

### **3.3. METADATOS**

Los metadatos y la ontología son las ideas fundamentales en las que se basa la representación de la información en la web semántica. Los metadatos se podrían definir como un conjunto de datos que describen el contenido informativo de un recurso, de archivos o de información. Un ejemplo para entender qué son los metadatos sería el catálogo de una base de datos o incluso el resumen de un documento, en el que se nos proporciona más información del documento que vamos a utilizar y de manera más accesible.

Relacionándolo con la web semántica, los metadatos son datos enormemente estructurados que describen aspectos de los recursos de información, tales como su calidad, detalles de los formatos, contenido, proceso de producción y otras características. Por lo que, los metadatos también se pueden definir como las descripciones estructuradas y codificadas que describen las características y propiedades de objetos y recursos para facilitar su recuperación, localización, administración, valoración, persistencia e interoperabilidad.

Se necesita aplicar junto a los metadatos un concepto adicional para poder realizar procesos más complejos con el fin de introducir en los recursos de información una estructura que permitan realizar deducciones a partir de axiomas que se aplican en los distintos dominios. Este concepto hace referencia a la ontología, puesto que permite expresar rasgos semánticos de los recursos.

### **3.4. ONTOLOGÍA**

#### **3.4.1. DEFINICIÓN DE ONTOLOGÍA**

La ontología tiene el objeto de detallar mejor un concepto, es decir, que viene a ser una organización bien estructurada de un concepto en la que aparte de poder almacenar información también la puede buscar y recuperar. Una ontología define las relaciones y los términos básicos para el entendimiento de un área del conocimiento, aparte de las reglas para aportar una definición más amplia a la hora de asociar los términos.

De lo que se trata es de convertir en conocimiento, a través de las ontologías que hacen referencia a los datos por medio de los metadatos mediante un esquema

común normalizado, un dominio del conocimiento. Aparte de que los metadatos detallan el esquema de datos presentes en las instancias también pueden acceder a informaciones con las que se puedan hacer deducciones, estableciendo axiomas que podrán aplicarse en los distintos dominios sobre el que trate el conocimiento guardado. Con esto los buscadores obtendrán información al compartir las estructuras de anotaciones web y los agentes del software, aparte de encontrar información precisa, podrán realizar interferencias de forma automática con la búsqueda de información relacionada con la que se encuentren en la página web y con las exigencias de las peticiones hechas por el usuario. Además, los creadores de páginas y servicios web podrán intercambiar y reutilizar datos siguiendo estas estructuras comunes.

La ontología se puede definir de distintas maneras:

- "Una ontología es un vocabulario acerca de un dominio: términos + relaciones + reglas de combinación para extender el vocabulario". Neches, 1991.
- "Una ontología es la especificación de una conceptualización". Gruber, 1993.
- "Una ontología es una especificación formal de una conceptualización compartida". Borst, 1997.
- "Una ontología es una base de datos que describe los conceptos generales o sobre un dominio, algunas de sus propiedades y cómo los conceptos se relacionan unos con otros". Weingand, 1997.
- "Una ontología necesariamente incluirá un vocabulario de términos y una especificación de su significado (definiciones e interrelaciones entre conceptos) que impone estructura al dominio y restringe las posibles interpretaciones". Uschold - Jasper.

Aunque la definición de Tom Gruber es la más utilizada en el área de la inteligencia artificial, "Una ontología es la especificación explícita de una conceptualización". Esta definición fue modificada por Borst quien lo define como: "Una ontología es una especificación formal de una conceptualización compartida". Studer y sus compañeros agregaron un significado más amplio a las definiciones de Gruber y Borst:

"Una ontología es una especificación explícita y formal sobre una conceptualización compartida".

- **Conceptualización:** La conceptualización hace referencia a un modelo abstracto de algún fenómeno en el mundo, ya identificados los conceptos relacionados con ese fenómeno.
- **Explícita:** Significa que el tipo de conceptos utilizados sean definidos explícitamente, esto quiere decir que, si además pueden definir otros conceptos del mismo tipo, se definan con detalle.
- **Formal:** Se refiere a que la ontología debe poder ser legible por la máquina, es decir, que se almacene en un formato digital.
- **Compartida:** Quiere decir que la ontología no sea restringida solo para un individuo, sino que pueda ser aceptada por un grupo de personas.

### 3.4.2. CARACTERÍSTICAS DE UNA ONTOLOGÍA

La ontología es un procedimiento en la que se representa un conocimiento resultado de elegir un dominio o ámbito del conocimiento, y aplicar sobre él un método para la obtención de una representación formal de los conceptos que contiene y de las relaciones existentes entre dichos conceptos. La ontología está formada por seis tipos de componentes que sirven para representar el conocimiento de algún dominio en específico: clases, atributos, relaciones, funciones, axiomas e instancias (Gruber, 1993):

- **Clases:** son los conceptos generales que se pretende concretar. También se conocen como conceptos y pueden tratar sobre cualquier dominio.
- **Relaciones:** van a representar las interacciones entre las distintas clases. Suelen formar la taxonomía del dominio.
- **Atributos:** representan la estructura interna de las clases, y pueden clasificarse en heredados y específicos. Los heredados vienen dados por las relaciones taxonómicas en las que la clase desarrolla el papel de hijo y hereda los atributos del padre, mientras que los específicos son aquellos que son propios de la clase a la que pertenecen.
- **Funciones:** representa un tipo de relación entre las clases del dominio.
- **Instancias:** representa objetos determinados de una clase. En las instancias, todos los atributos de las clases tienen un valor concreto.
- **Reglas de restricción o axiomas:** se tratan de teoremas que se le aplica a la ontología para que sus elementos cumplan las relaciones impuestas, es decir, que aplican la lógica (Si X y Z son de la clase Y, entonces X no es subclase de Z”).

Algunas de las características que poseen las ontologías son:

- **Ontologías múltiples:** en algunas ocasiones se necesita combinar dos o más ontologías para definir una ontología cuyo propósito es hacer explícito algún punto de vista. En ese caso, cada ontología introduce conceptualizaciones específicas.
- **Multiplicidad de la representación:** una clase puede ser representada de distintas formas, por lo que pueden existir varias representaciones de la misma clase.
- **Mapeo de ontologías:** entre los elementos de una o más ontologías se podría establecer las relaciones con el fin de establecer especializaciones, generalizaciones, conexiones, etc.
- **Distintos niveles de abstracción de las ontologías:** Estos niveles de abstracción o generalización nos da la tipología de las ontologías. La idea es caracterizar una red de ontología con el uso de abstracción y multiplicidad.

Algunas propiedades que deben cumplir las ontologías son:

- **Coherencia:** para advertir en el caso de que haya consistencias en las definiciones.
- **Claridad:** la ontología debe poder comunicar de forma efectiva el significado de sus términos. Las definiciones deben ser lo más objetivas posibles y deben explicarse en lenguaje natural.

- **Sesgo de codificación mínimo:** no debe de depender de una codificación particular a nivel de símbolo para la especificación del conocimiento.
- **Extensibilidad:** deben anticiparse nuevos usos de vocabularios compartidos.
- **Compromiso ontológico:** debe aportar la menor cantidad de supuestos acerca del mundo modelado, para así tener una ontología más objetiva y formal.

### 3.5. TECNOLOGÍA UTILIZADA EN LA WEB SEMÁNTICA

Para hacer posible la Web Semántica, se ha creado una tecnología que incorpora lenguajes para dar forma a las ontologías, lenguajes de consulta, módulos de gestión, de ontologías, conversión de ontologías, módulos de visualización, entornos de desarrollo, y otras herramientas y librerías. El primer lenguaje para estructurar la web semántica fue el SHOE, creado por Jim Hendler en la Universidad de Maryland en 1997. Desde entonces han surgido otros lenguajes y estándares con un fin parecido.

A Continuación, se detallan las tecnologías más significativas: XML, RDF, SPARQL, y más recientemente OWL.

- **XML:** Es el acrónimo de Extensible Markup Language (Lenguaje Extensible de Marcación), lo que quiere decir que, es un lenguaje de marcado que define un conjunto de reglas para la codificación de documentos. Es un lenguaje ampliamente utilizado en el almacenamiento, estructuración e intercambio de una gran cantidad de datos entre sistemas de información. Y su principal función es la de describir los datos en lugar de mostrarlos.
- **RDF:** Es el acrónimo de Resource Description Framework. Fue creada en 1998 y recomendada por W3C en 1999, su función es la de describir mediante grafos y etiquetados los recursos, para facilitar su integración desde diferentes fuentes de datos.
- **SPARQL:** Es el acrónimo de SPARQL Protocol and RDF Query Language. Es un lenguaje de recuperación que permite realizar consultas acerca de los datos en un grafo de RDF.
- **OWL:** Es el acrónimo de Web Ontology Language. Es un lenguaje de ontología desarrollada por W3C para la definición de términos y relaciones de un vocabulario de la web. El OWL es una extensión del RDF por lo que tiene mayor capacidad para expresar significado y semántica de manera que puede ser razonado por programas de computadoras que verifican la coherencia del conocimiento permitiendo que conocimiento implícito se convierta en conocimiento explícito. Este lenguaje es el que se usará en el programa Protégé, para la realización de este proyecto.

Según Berners-Lee, la arquitectura de la Web Semántica se podría representar de la siguiente forma:

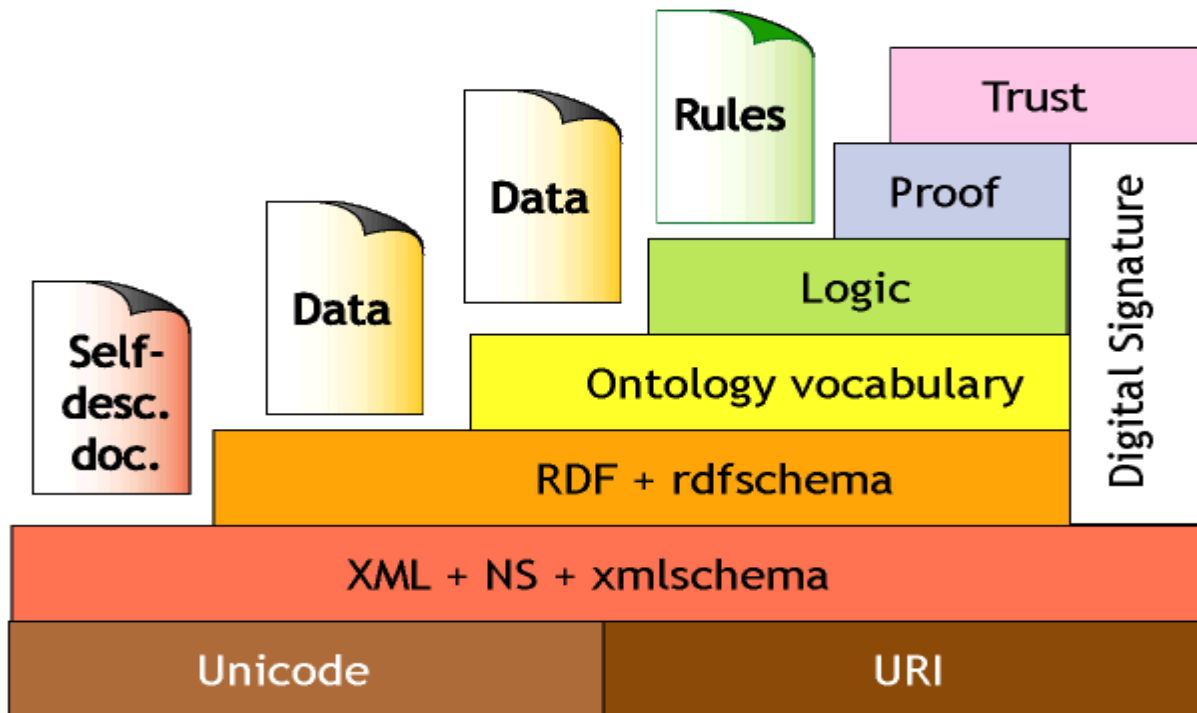


Figura 1: Representación de la arquitectura de la web semántica <sup>23</sup>

## 4. PROGRAMA Y LENGUAJE UTILIZADOS PARA LA ONTOLOGÍA

### 4.1. PROTÉGÉ



Figura 2: Logo del programa de Protégé <sup>25</sup>

El Protégé es el programa utilizado para la construcción de la ontología de este proyecto. Esta es una versión importante de Protégé que incorpora varias mejoras en la interfaz de usuario y corrige muchos errores. Consultar <https://github.com/protegeproject/protege-distribution/releases/tag/v5.5.0> para más información acerca de las mejoras. A continuación, se detallan algunas de las características del programa.

<b>Nombre de la herramienta de creación de ontologías</b> <i>Protégé 5</i>
<b>Versión y fecha de lanzamiento de la versión</b> <i>Release 5.5.0. 15 de marzo de 2019</i>
<b>Licencia</b> <i>BSD 2 cláusula</i>
<b>Desarrolladores (organización a la que pertenece)</b> <i>Centro de stanford para la investigación en informática Biomédica</i>
<b>Información complementaria (donde se puede encontrar)</b> <i><a href="https://protege.stanford.edu/">https://protege.stanford.edu/</a></i>
<b>Descripción</b> <i>Protégé es una plataforma gratuita de código abierto que proporciona a una comunidad de usuarios en crecimiento un conjunto de herramientas para construir modelos de dominio y aplicaciones basadas en el conocimiento con ontologías.</i>

## 4.2. LENGUAJE OWL

El Lenguaje de Ontología Web (OWL) desarrollada por la W3C, es un lenguaje de la Web Semántica diseñado para representar un conocimiento rico y complejo sobre cosas, grupos de cosas y relaciones existentes entre cosas. OWL es un lenguaje basado en la lógica computacional, de forma que el conocimiento expresado en OWL puede ser explotado por programas de computadora, por ejemplo, para verificar la consistencia de ese conocimiento o para hacer explícito el conocimiento implícito. Los documentos de OWL, son conocidos como ontologías, se pueden publicar en la World Wide Web y pueden hacer referencia a otras ontologías de OWL o desde ellas. OWL es parte de la tecnología de Web Semántica, que incluye RDF, RDFS, SPARQL, etc.

La versión actual de OWL, conocida como "OWL 2", fue desarrollada por el W3C OWL Working Group y publicada en 2009, con una segunda edición publicada en 2012. OWL 2 es una extensión y revisión de la versión 2004 de OWL y publicada en 2004.

OWL dispone de tres lenguajes, los cuales van de menor a mayor en cuanto al nivel expresividad, y está diseñado para utilizarse por cualquier usuario capaz de manejar los distintos niveles de lenguajes.

- OWL Lite:** está diseñado para los usuarios inexpertos que solo quieren clasificar los conceptos y utilizar restricciones simples. Un ejemplo es que sólo permite establecer valores cardinales de 0 o 1 al mismo tiempo que admite restricciones de cardinalidad. Es el lenguaje más sencillo puesto a su facilidad de implementación y comprensión. OWL Lite proporciona una salida para tesauros y otras taxonomías. OWL Lite tiene una complejidad formal menor que la OWL DL.

- **OWL DL:** diseñado para aquellos usuarios que quieren una mayor expresividad que el OWL lite, pero con la conservación de toda la parte computacional. OWL DL posee todas las construcciones del lenguaje de OWL, y solo se usarán bajo ciertas restricciones. OWL DL es el acrónimo de (Description Logics, en inglés) y es denominada así por su relación con la lógica de descripción, un campo de investigación que estudia la lógica que compone la base formal de OWL.
- **OWL Full:** diseñado para usuarios expertos que quieren la mayor expresividad y libertad sintáctica de RDF y a diferencia del OWL DL no posee garantías computacionales. OWL Full permite a la ontología aumentar el significado del vocabulario preestablecido (RDF o OWL).

Para la ontología de este proyecto, al utilizar la versión más reciente del programa, este lleva incorporado el lenguaje de OWL DL, y es el lenguaje OWL que se utiliza, pudiendo guardar la ontología en un formato OWL/XML.

## 5. LA INSTALACIÓN ELÉCTRICA Y SUS COMPONENTES SEGÚN LA REBT

### 5.1. COMPONENTES DE UNA INSTALACIÓN ELÉCTRICA

Una instalación eléctrica está compuesta por los siguientes elementos:

- **Línea de acometida:** Es la línea que parte de la red de distribución y alimenta la Caja General de Protección. Las acometidas pueden ser aéreas, subterráneas, o mixtas en función del sistema de instalación y depende también de la red de distribución a la cual se conectan. Esta línea pertenece a la empresa suministradora, y se compone de 3 cables conductores de fase y 1 cable del neutro (trifásica).
- **Caja general de protección (CGP):** La Caja General de Protección almacena los elementos de protección de la línea general de alimentación. Dispone de tres fusibles en su interior (uno por cada conductor de fase) que protegen contra posibles cortocircuitos. La CGP se suelen localizar en la fachada, u otros lugares de zona comunes con libre acceso.
- **Línea general de alimentación (LGA):** La Línea Repartidora o Línea General de Alimentación conecta la CGP con la centralización de contadores. Está compuesto de 4 cables, los cuales 3 son de fase y 1 de neutro. Y por supuesto, el cable de protección (toma de tierra).
- **Centralización de contadores (CC):** en las concentraciones de contadores se albergan todos los aparatos de mando, medida, control y protección de las derivaciones individuales que se alimentan desde dicha concentración. El contador es un aparato que se encarga de la medición y el registro del consumo de energía eléctrica. Suele haber un contador por usuario o vivienda, pero en el caso de edificios con muchos usuarios, todos se encuentran en un espacio común (armario, recinto, habitación) que se denomina centralización de contadores. La centralización de contadores está formada por los siguientes elementos:
  - **Interruptor general de maniobra:** interruptor encargado de la desconexión de toda la centralización mediante el corte de la corriente que llega desde la Línea Repartidora a la concentración de contadores.

- **Unidad de embarrado general y fusibles de seguridad:** se tratan de cuatro barras metálicas conectadas a los cuatro conductores de la LGA (3 fases + 1 neutro). Los cables eléctricos salen de los embarrados y se conectan a los contadores a través de fusibles de seguridad, es decir, que cada contador debe tener su propio fusible de seguridad.
- **Unidad de medida:** Está formado por los contadores que se encargan del control del consumo eléctrico de los usuarios además de interruptores horarios y dispositivos de mando.
- **Derivaciones Individuales y embarrado de protección:** Las Derivaciones Individuales salen de los contadores y se conectan con las instalaciones de los usuarios. El embarrado de protección está compuesto de por unas barras metálicas conectadas a tierra y a su vez a los cables de tierra de cada Derivación Individual.
- **Derivaciones individuales (DI):** Las derivaciones individuales derivan del contador de los usuarios y transportan la energía eléctrica al Interruptor de Control de Potencia, que se encuentran en las viviendas, aunque a veces este interruptor suele estar integrado en el contador. Cada derivación individual está compuesta por 3 cables uno de fase, uno neutro y otro de tierra. Esto quiere decir que, el suministro a los usuarios se realiza en monofásica.
- **Interruptor de control de potencia (ICP):** El Interruptor de Control de Potencia puede ubicarse en el cuadro general de mando y protección o integrado en los contadores y propiedad de la empresa suministradora. Se encarga de limitar el consumo eléctrico cortando la energía en el caso de que el usuario supere la potencia contratada. El ICP suele ubicarse en el Cuadro General de Mando y Protección o integrado en los contadores.
- **Dispositivos Generales de Mando y Protección (CGMP):** Se situarán lo más próximo al punto de entrada de la derivación individual en el local o vivienda del usuario. Se ubican en el interior del cuadro general de mando y protección (CGMP), y se compone de:
  - **Interruptor de Control de Potencia (ICP).**
  - **Interruptor General (IG):** Es un interruptor magnetotérmico que se encarga de proteger las instalaciones interiores frente a sobrecargas o cortocircuitos. El Interruptor General corta automáticamente la corriente cuando detecta un gran aumento en la intensidad de corriente que circula.
  - **Interruptor Diferencial (ID):** Interruptor encargado de la protección frente a posibles contactos indirectos de todos los circuitos cuando se produce alguna fuga de corriente en la instalación.
  - **Pequeños Interruptores Automáticos (PIAs):** Los PIAs son interruptores automáticos magnetotérmicos encargados de proteger cada circuito eléctrico independiente de la vivienda o local, frente a fallos en la instalación.
- **Toma de tierra:** La toma de tierra consiste en una pica o electrodo enterrado a tierra, a la que se le conecta un cable (color verde amarillo) conectado a todos los aparatos eléctricos de las viviendas o del edificio. Su función es la de llevar a tierra cualquier fuga de corriente para así proteger al usuario en el caso de contacto directo con cualquier aparato cargado eléctricamente



## 5.2. INSTALACIÓN ELÉCTRICA REALIZADO EN PROTÉGÉ PARA EL PROYECTO

Para comprobar si funciona la ontología que se ha realizado en el programa, se hará la instalación eléctrica de una casa de cuatro viviendas. Esta casa se ubica en Calle Princesa Guajara. Código postal 38611. San Isidro. S/C de Tenerife.



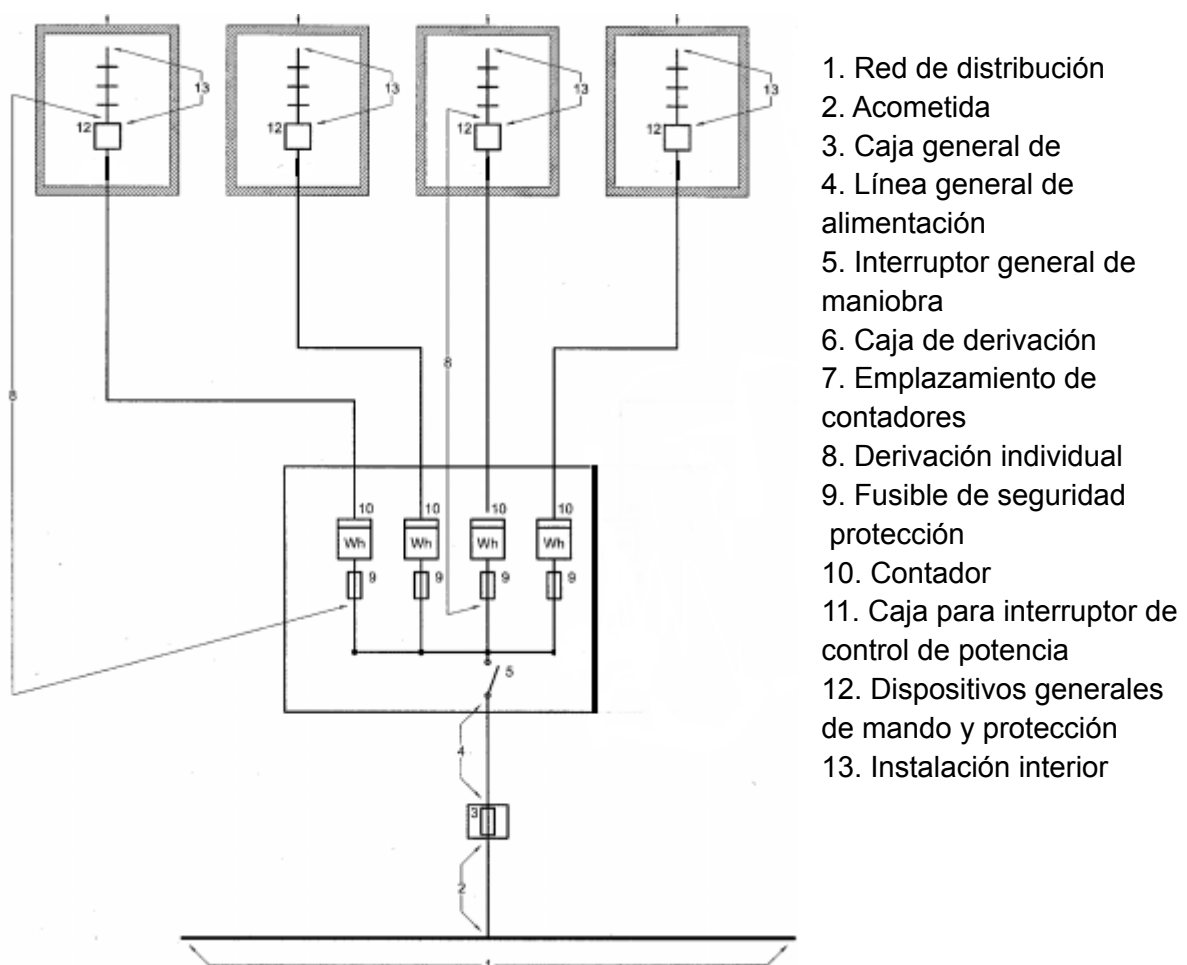
*Figura 3: Casa del que se hace la instalación eléctrica*

### 5.2.1. COMPONENTES DE LA INSTALACIÓN ELÉCTRICA

- **Línea de acometida:** se trata de una acometida subterránea con un sistema de instalación de entrada y salida. Utiliza cables aislados con tensión asignada de 0,6/1 kV con una longitud que oscila entre los 12 y 14 metros. Este tipo de instalación se hará conforme a la ITC-BT-07.
- **Caja general de protección:** se ha instalado en un nicho en la pared como se muestra en la figura 3, a una altura de 0,4 metros del suelo y la puerta con grado de protección IK10 según UNE-EN 50.102.

- **Línea general de alimentación:** está constituido por conductores aislados en el interior de tubos enterrados.
- **Centralización de contadores:** aquí se encuentran los cuatro contadores para cada vivienda ubicados en el interior de un armario con grado de protección IK 09 y IP40. Estos contadores llevan integrados el interruptor de control de potencia por lo que no hará falta instalarlo en el cuadro general de mando y protección.
- **Derivación individual:** la casa consta de 4 derivaciones individuales, una para cada vivienda. Está constituida por conductores aislados en el interior de canales protectores que no se pueden acceder sin ayuda de un profesional. Los conductores utilizados son de aluminio, aislados y unipolares, con tensión asignada de 450/750V.
- **Interruptor general de maniobra:** se ubica entre la línea general de alimentación y la centralización de contadores, con una intensidad de 160 A.
- **Cuadro general de mando y protección:** ubicado el lado de la puerta de entrada de las viviendas a una altura de 1,6 metros y con grado de protección IP30.
- **Dispositivos generales de mando y protección:** para cada vivienda se ha instalado un interruptor diferencial de 25 A con sensibilidad de 300mA, un interruptor automático 25 A y 5 interruptores magnetotérmicos (de 10, 2x16, 20 y 25 amperios).
- **Toma a tierra:** Cable LEXMAN H07V-K verde/amarillo 2,5 mm<sup>2</sup> 10 m.

El esquema unifilar de dicha instalación quedaría de la siguiente manera:



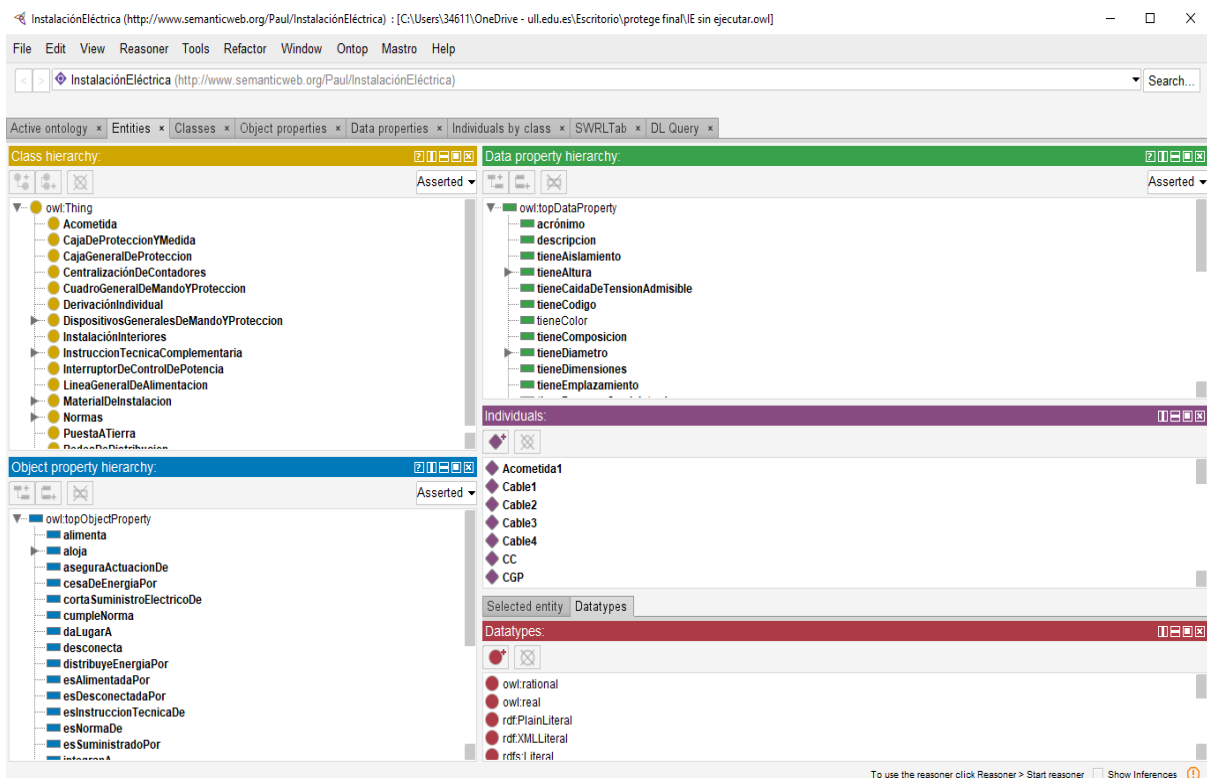
**Figura 4:** Esquema unifilar de la instalación eléctrica adaptada del REBT <sup>5</sup>

## 6. COMPONENTES DE ONTOLOGÍA OWL UTILIZADAS EN EL PROGRAMA PARA LA CONSTRUCCIÓN DE LA ONTOLOGÍA

Para realizar la ontología de la instalación eléctrica en este proyecto, se ha basado siempre en el reglamento electrotécnico para baja tensión (REBT).

A continuación, se detallan los componentes más utilizados en el programa durante la realización del proyecto. Estos componentes se pueden separar en pestañas o también se pueden englobar en una ventana, como vemos en la siguiente imagen.

**Razonador**  
**Clases**  
**Object properties**  
**Data properties**  
**Individuals**  
**SWRL**



**Figura 5:** Componentes de la ontología en Protégé

Estos componentes ya han sido mencionados anteriormente en el punto de las características de una ontología (apartado 4.4.2), aunque en el programa varía el nombre.

- Clases, se traduce como Clases en el programa
- Relaciones, se traduce como object properties en el programa

- Atributos, se traduce como data properties
- Instancias se traduce como individuos
- Por último, las reglas se realizan con el lenguaje SWRL (semantic web rule language)

## 6.1. CLASES

La clase es el componente principal de la ontología, en el que se definen los distintos conceptos. Se quiere realizar la ontología de una instalación eléctrica por lo que estos conceptos deben ser de dicha instalación.

Las clases implementadas para la ontología de una instalación eléctrica son sacadas del Reglamento electrotécnico de baja tensión detallados en el punto 5 y se muestran a continuación:

<b>Clases</b>	<b>Subclases</b>	<b>Disjunto</b>
Acometida		Disjunta con todas las otras clases
Caja de protección y medida		Disjunta de todas las otras clases menos de materiales de instalación
Caja general de protección		Disjunta de todas las otras clases menos de materiales de instalación
Centralización de contadores		Disjunta con todas las otras clases
Cuadro general de mando y protección		Disjunta de todas las otras clases menos de materiales de instalación
Derivación individual		Disjunta con todas las otras clases
Dispositivo general de mando y protección	<ul style="list-style-type: none"> <li>- Interruptor diferencial</li> <li>- Interruptor general automático</li> <li>- Pequeños interruptores automáticos</li> </ul>	Disjunta de todas las otras clases menos de materiales de instalación
Instalaciones interiores		Disjunta con todas las otras clases
Instrucción técnica complementaria	<ul style="list-style-type: none"> <li>- Instrucción técnica para Acometida</li> <li>- Instrucción técnica para cajas generales de protección</li> <li>- Instrucción técnica para derivación individual</li> </ul>	Disjunta con todas las otras clases

	<ul style="list-style-type: none"> <li>- Instrucción técnica para DGMP y ICP</li> <li>- Instrucción técnica para instalación interior</li> <li>- Instrucción técnica para línea general de alimentación</li> <li>- Instrucción técnica para puesta a tierra</li> <li>- Instrucción técnica para red de distribución</li> </ul>	
Interruptor de control de potencia		Disjunta de todas las otras clases menos de materiales de instalación
Línea general de alimentación		Disjunta con todas las otras clases
Material de instalación	<ul style="list-style-type: none"> <li>- Alumbrado</li> <li>- Apoyos</li> <li>- Cables</li> <li>- Cajas de empalme</li> <li>- Canalizaciones</li> <li>- Cuadros de maniobra</li> <li>- Cuadros eléctricos</li> <li>- Detectores</li> <li>- Fusibles de seguridad</li> <li>- Interruptores</li> <li>- Temporizadores</li> <li>- Otros materiales</li> </ul>	Disjunta de todas las otras clases menos de las cajas de protección, el cuadro, el interruptor de control de potencia y los dispositivos generales de mando y protección
Normas	<ul style="list-style-type: none"> <li>- Normas para canalizaciones</li> <li>- Normas para conductores</li> <li>- Normas para cuadros</li> </ul>	Disjunta con todas las otras clases
Puesta a tierra		Disjunta con todas las otras clases
Red de distribución		Disjunta con todas las otras clases

La palabra disjunta (Disjoint with en Protégé) significa que no tiene ningún elemento en común con otro conjunto, por lo que, traducido en el programa, significa que un mismo individuo no puede pertenecer a dos clases disjuntas. Como se aprecia en la tabla, las clases están todas disjuntas, menos algunas que no lo están de materiales de instalación. Por ejemplo, la caja general de protección no está disjunta de materiales de instalación, por lo que un individuo puede ser una caja de protección y también un material de instalación. En cambio, un individuo perteneciente a

instrucción técnica no puede ser un material de instalación ni una acometida, por lo que estas están disjuntas.

A continuación, se aprecia en la imagen cómo quedarían las clases implementadas para la ontología en Protégé.

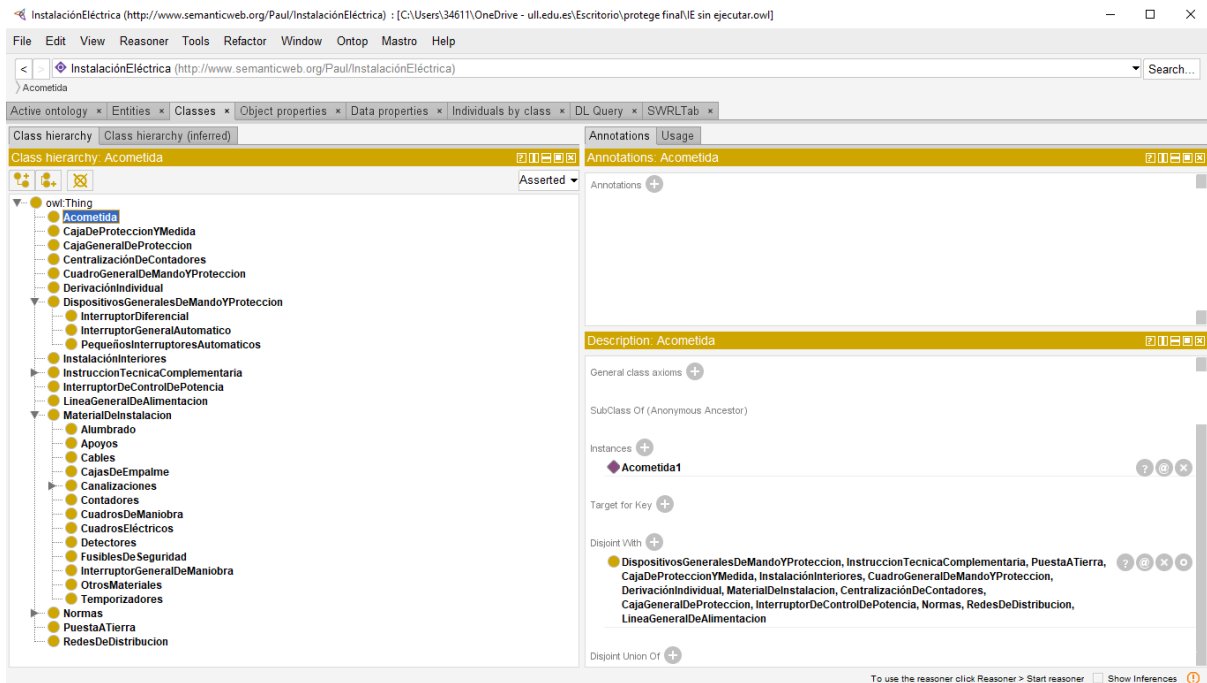


Figura 6: Clases de la ontología

## 6.2. PROPIEDADES DE OWL

Las propiedades OWL representan relaciones. Hay dos tipos de propiedades, propiedades de objeto (object properties) y propiedades de datos (data properties).

Cabe destacar que a pesar de que no es obligatorio, se recomienda que los nombres de las propiedades empiecen por una letra minúscula, sin espacios y con las demás palabras con letra mayúsculas. También es recomendada empezar las propiedades con verbos como, “tiene”, “es” etc, por ejemplo, esNormaDe, estaAlimentadaPor, tieneSección. Este formato ayudará a que las propiedades sean más claras a la hora de entenderlo.

En la descripción de las propiedades de OWL, debe haber un dominio y un rango de manera que la estructura quedaría así:

“DOMINIO” -----> “PROPIEDADES” -----> “RANGO”

Ejemplo:

- En el caso de las propiedades de objeto:

“Caja general de protección”

“esAlimentadaPor”

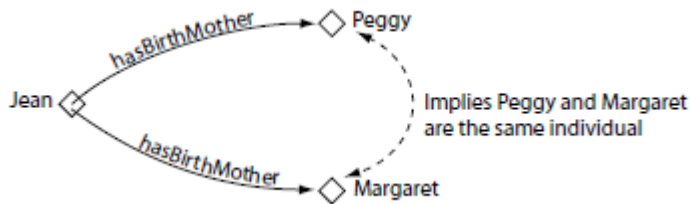
“Acometida”

- En el caso de las propiedades de datos  
 "Cable"            "tieneSección"            "16mm2"

### 6.2.1. CARACTERÍSTICAS DE LAS PROPIEDADES

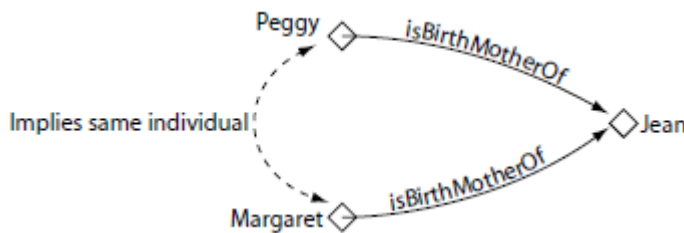
En el programa Protégé, las características que pueden adoptar las propiedades de objeto son.

- **Funcional:** si una propiedad es funcional, para un cierto individuo, solo puede haber un individuo relacionado con ese individuo a través de dicha propiedad.



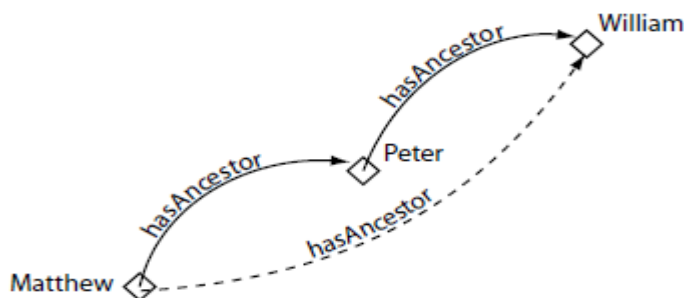
**Figura 7.1:** Ejemplo de una propiedad funcional <sup>24</sup>

- **Funcional Inversa:** si una propiedad es funcional inversa, esto significa que su propiedad inversa, es funcional.



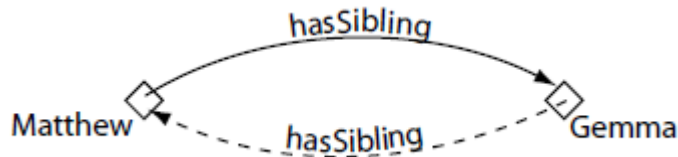
**Figura 7.2:** Ejemplo de una propiedad funcional inversa <sup>24</sup>

- **Transitiva:** si una propiedad es transitiva, esto significa que, si una propiedad relaciona un individuo A al individuo B, y la misma propiedad relaciona el individuo B al individuo C, entonces el individuo A se relaciona al individuo C, a través de dicha propiedad.



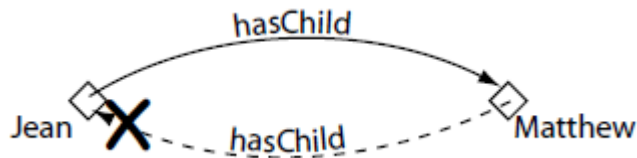
**Figura 7.3:** Ejemplo de una propiedad transitiva <sup>24</sup>

- **Simétrica:** si una propiedad es simétrica, quiere decir que dicha propiedad relaciona a un individuo A al individuo B, y también esa misma propiedad relaciona el individuo B al individuo A.



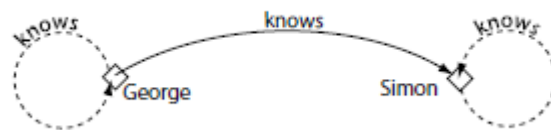
**Figura 7.4:** Ejemplo de una propiedad simétrica <sup>24</sup>

- **Asimétrica:** si una propiedad es asimétrica, quiere decir que dicha propiedad relaciona a un individuo A al individuo B, pero esa misma propiedad no puede relacionar el individuo B al individuo A.



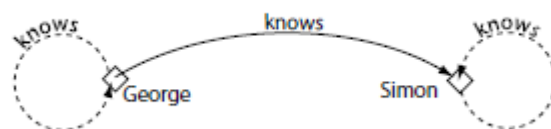
**Figura 7.5:** Ejemplo de una propiedad asimétrica <sup>24</sup>

- **Reflexiva:** si una propiedad es reflexiva, quiere decir que dicha propiedad puede relacionar un individuo A, a sí mismo.



**Figura 7.6:** Ejemplo de una propiedad reflexiva <sup>24</sup>

- **Irreflexiva:** si una propiedad es irreflexiva, quiere decir que dicha propiedad no puede relacionar a un individuo A, a sí mismo.



**Figura 7.7:** Ejemplo de una propiedad irreflexiva <sup>24</sup>

A continuación, se muestra la imagen de cómo son las características en el programa Protégé.



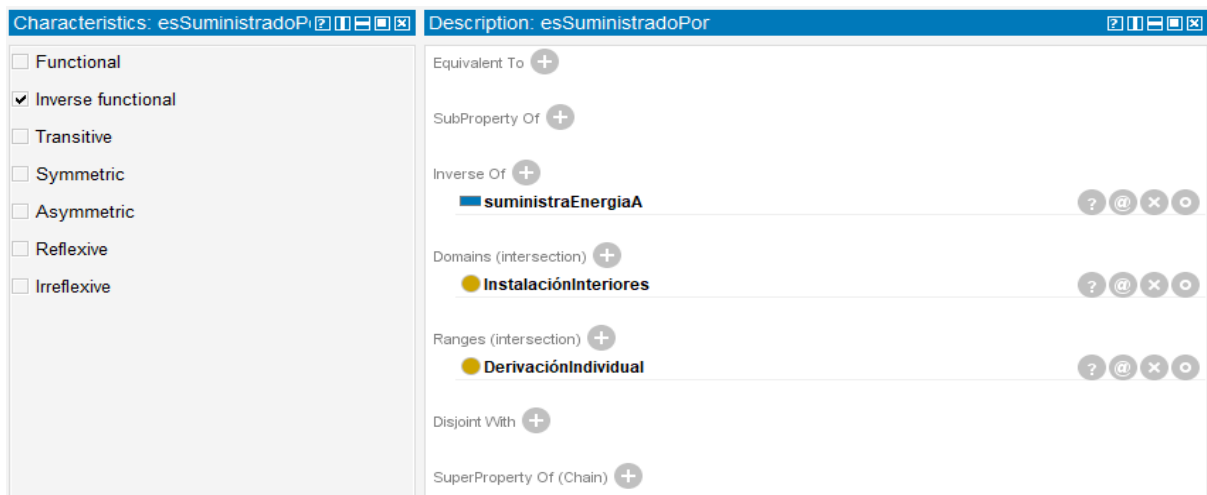


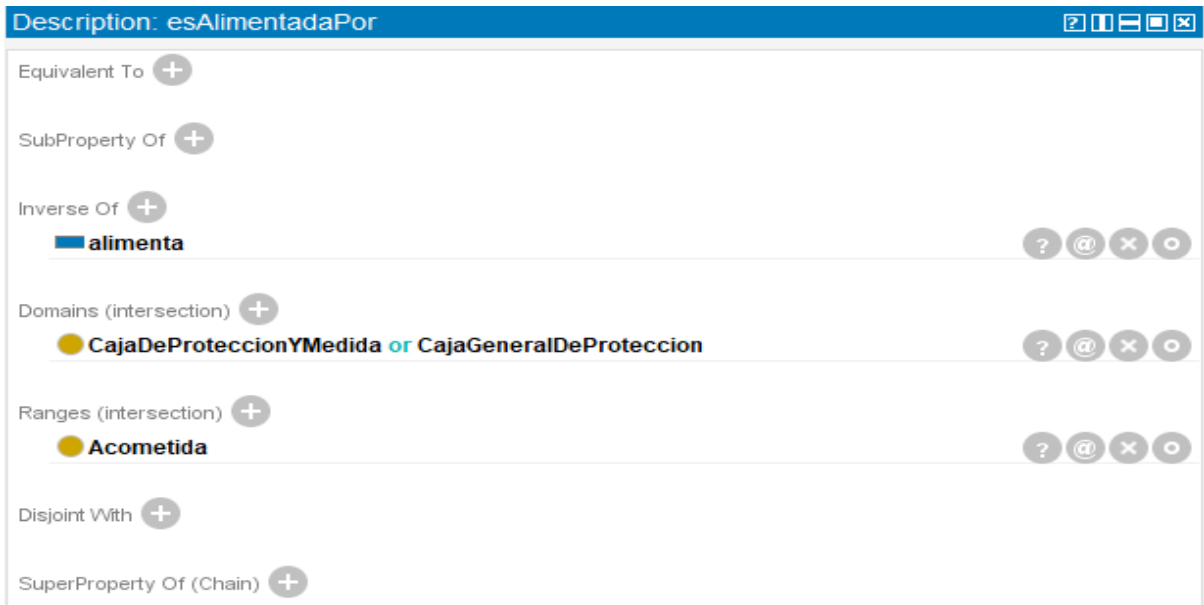
Figura 7: Características de las propiedades en Protégé

## 6.2.2. PROPIEDADES DE OBJETOS

Las propiedades de objeto (object properties) son aquellas que relacionan dos clases entre sí, es decir, que se tendrá como dominio y rango de una propiedad de objeto la misma o dos clases distintas. Se podría indicar más de un dominio y/o rango, pero esta sería la intersección entre estos.

También se puede definir la inversa de una propiedad de objeto cuyos rangos y dominios se heredan de su propiedad inversa. Pero al ser la inversa, el dominio de uno se convierte en el rango del otro, y el rango en el dominio. En las siguientes imágenes vemos un ejemplo de dos propiedades inversas implementadas en el programa.





**Figuras 8:** Ejemplo de Propiedades inversas. Alimenta; es alimentada por

A partir de ahora se utilizarán acrónimos para definir algunas clases. Los acrónimos se muestran a continuación:

Nombre completo	Acrónimo
Caja general de protección	CGP
Centralización de contadores	CC
Caja de protección y medida	CPM
Derivación individual	DI
Dispositivos generales de mando y protección	DGMP
Instrucción técnica complementaria	ITC
Interruptor de control de potencia	ICP
Interruptor diferencial	ID
Interruptor general automático	IGA
Interruptor general de maniobra	IGM
Línea general de alimentación	LGA
Fusible de seguridad	FS
Pequeños interruptores automáticos	PIAs

Las propiedades de objetos implementadas en el programa para la Instalación Eléctrica fueron sacadas de la REBT para enlazar las diferentes clases y detallan a continuación:

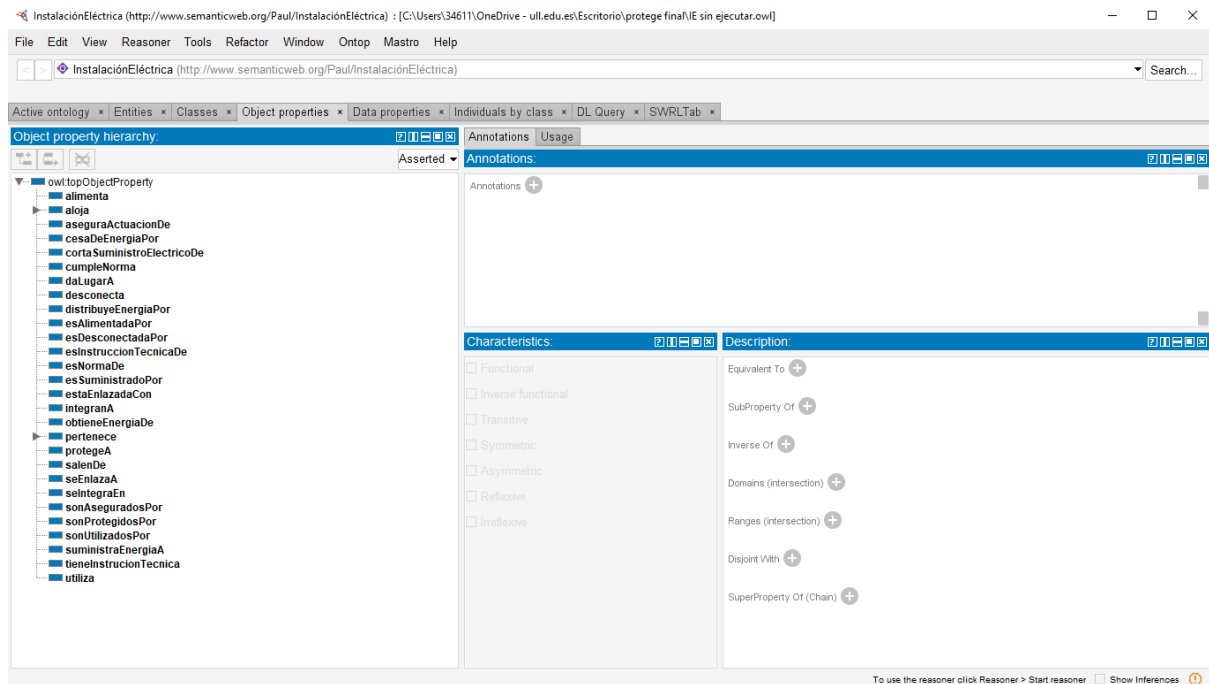
<b>Propiedades de objeto</b>	<b>Inversa</b>	<b>Dominio</b>	<b>Rango</b>
alimenta	esAlimentadaPor	Acometida	CGP o CPM
esAlimentadaPor	alimenta	CGP o CPM	Acometida
acomoda	sonAcomodadosEn	CGP	Fusibles de seguridad
sonAcomodadosEn	acomoda	Fusibles de seguridad	CGP
alberga	seUbicaEn	CMP	Contadores o fusibles de seguridad
seUbicaEn	alberga	Contadores o fusibles de seguridad	CPM
alojaA	sonInstaladosEn	CGMP	DGMP o ICP
sonInstaladosEn	alojaA	DGMP o ICP	CGMP
tienenAlbergados	seAlberganEn	CC	Contadores o DI o FS o IGM
seAlberganEn	tienenAlbergados	Contadores o DI o FS o IGM	CC
aseguranActuación De	sonAseguradosPor	Puesta a tierra	DGMP

<b>Propiedades de objeto</b>	<b>Inversa</b>	<b>Dominio</b>	<b>Rango</b>
sonAseguradosPor	aseguranActuación De	DGMP	Puesta a tierra
cesaDeEnergíaPor	cortaSuministroEléctricoDe	Instalación interior	ICP
cortaSuministroEléctricoDe	cesaDeEnergíaPor	ICP	Instalación interior
cumpleNorma	esNormaDe	CPM o CGP o CGMP o DGMP o	Norma

		Material de instalación	
esNormaDe	cumpleNorma	Norma	CPM o CGP o CGMP o DGMP o Material de instalación
daLugarA	salenDe	Contadores	DI
salenDe	daLugarA	DI	Contadores
desconecta	esDesconectadaPor	IGM	CC
esDesconectadaPor	desconecta	CC	IGM
distribuyeEnergíaPor	obtieneEnergíaDe	Red de distribución	Acometida
obtieneEnergíaDe	distribuyeEnergíaPor	Acometida	Red de distribución
esInstrucciónTécnicaDe	tieneInstrucciónTécnica	ITC	Todas las clases
tieneInstrucciónTécnica	esInstrucciónTécnicaDe	Todas las clases	ITC
esSuministradoPor	suministraEnergíaA	Instalación interior	DI
suministraEnergíaA	esSuministradoPor	DI	Instalación interior
integranA	seIntegranEn	Contadores	ICP
seIntegranEn	integranA	ICP	Contadores
seComponeDe	sonComponentesDe	DGMP	ID o IGA o PIAs
sonComponentesDe	seComponeDe	ID o IGA o PIAs	DGMP
protegeA	sonProtegidosPor	DGMP	Instalación interior
sonProtegidosPor	protegeA	Instalación interior	DGMP
sonUtilizadosPor	utiliza	Cables	Acometida o DI o LGA o red de distribución
utiliza	sonUtilizadosPor	Acometida o DI o LGA o red de distribución	Cables
estáConectadoCon		Contador	FS

estáEnlazadaCon	seEnlazaA	CGP	LGA
seEnlazaA	estáEnlazadaCon	LGA	CGP

A continuación, se aprecia en la imagen cómo quedarían las propiedades de objeto implementadas para la ontología en Protégé.



**Figura 9:** Propiedades de objeto implementadas en el programa

En cuanto a las características de las propiedades de objeto, sólo se ha encontrado las funcionales y las simétricas:

Características	Propiedad de objeto
Funcional	alojaA, cesaDeEnergiaPor, deLugarA, desconecta, integranA, protegeA, estaConectadoA, suministraEnergiaA, estaEnlazadaA
Funcional inversa	cortaSuministroElectrico, esDesconectadaPor, esSuministradoPor, sonInstaladosEn, salenDe, seIntegranEn, sonProtegidosPor, seEnlazaA
Simétrica	estaConectadoA

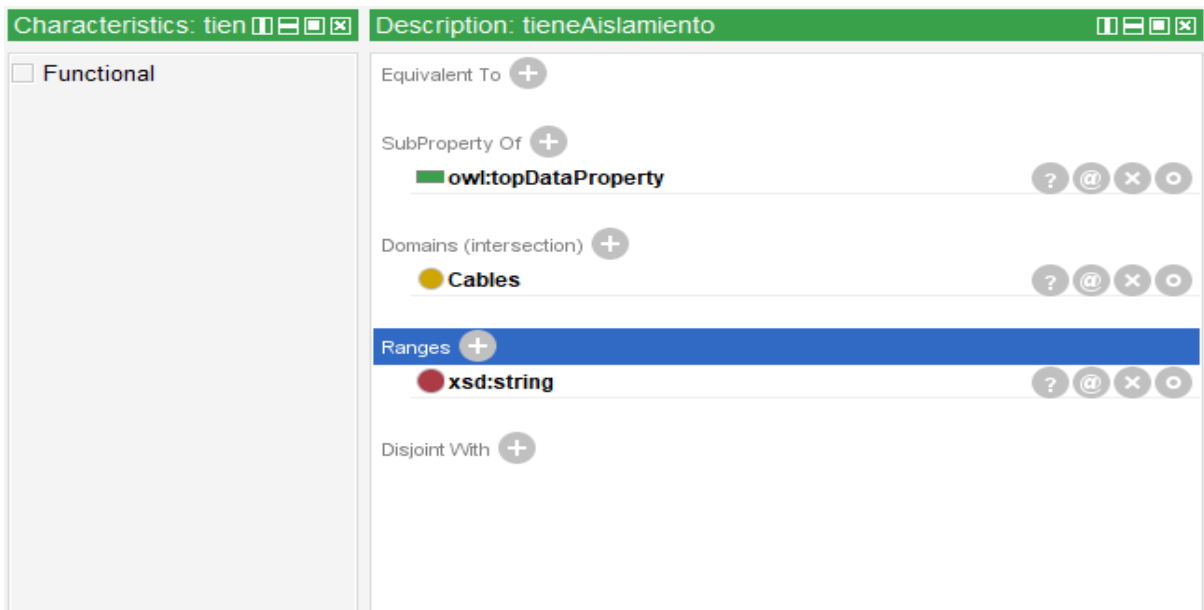
### 6.2.3. PROPIEDADES DE DATOS

Las propiedades de datos (data properties) son propiedades que definen las características de una o varias clases. Se diferencia de la propiedad de objeto en

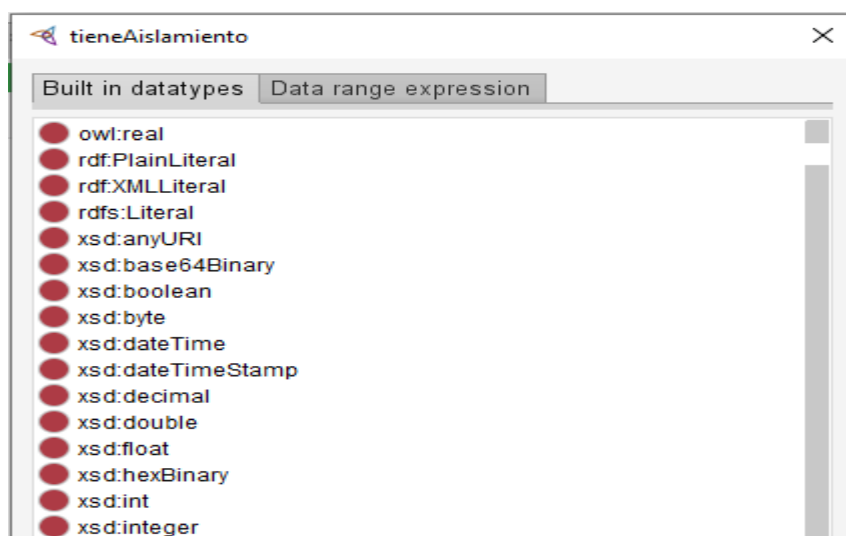
que, el rango ya no es una clase sino un literal. Estos literales pueden ser de tipo numérico o de tipo texto.

La única característica que posee las propiedades de datos en el programa de Protégé es la funcional, aunque no se ha seleccionado por el hecho de que se asume que los individuos sólo se relacionarán a través de una sola propiedad de datos, es decir que no se utilizará más de una vez la misma propiedad para definir dicha característica (la característica es el rango y este debe ser un literal).

En las siguientes imágenes se puede ver cómo es la descripción de las propiedades de datos en el programa y algunos literales que se podrían utilizar a la hora de definir el rango de una propiedad.



**Figura 10:** Dominio y rango de las propiedades de dato



**Figura 11:** Literales del rango de las propiedades de dato

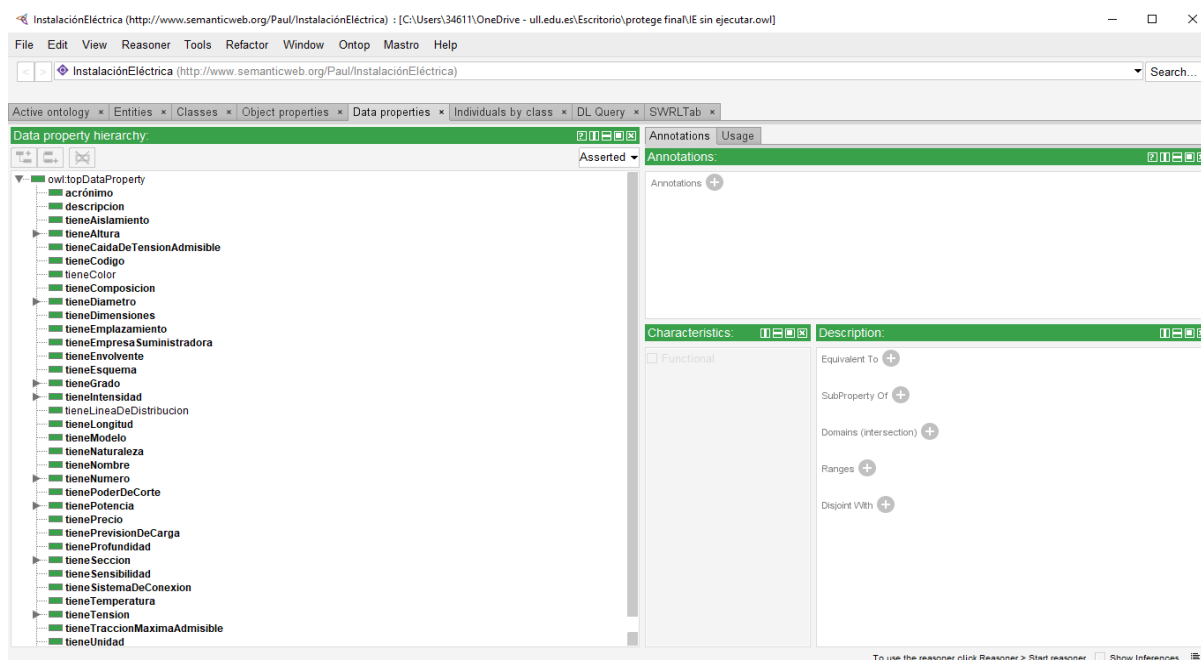
Las propiedades de datos implementadas en el programa para la Instalación Eléctrica se detallan a continuación:

<b>Propiedades de datos</b>	<b>Dominio</b>	<b>Rango</b>
acrónimo	Todos los acrónimos descritos anteriormente	xsd:string
descripción	Todas las clases en el programa	xsd:string
tieneAislamiento	Cables	xsd:string
tieneAltura	Acometida o CGP o CPM o CGMP o DGMP	xsd:decimal
tieneCaídaDeTensión Admisible	Acometida o LGA	xsd:float
tieneCódigo		xsd:string
tieneColor	Puesta a tierra	xsd:string
tieneComposición	Acometida o red de distribución o DI o LGA	xsd:string
tieneDiámetro	Cables o canalizaciones	xsd:float
tieneDimensiones	Cables o canalizaciones o CGP o CPM o CGMP	xsd:string
tieneEmplazamiento	Acometida o CPM o CGP o contadores o CGMP o DGMP o FS o ICP	xsd:string
tieneEmpresa Suministradora	Red de distribución	xsd:string
tieneEnvolvente	CPM o CGP o CGMP	xsd:string
tieneEsquema		xsd:string
tieneGrado	CPM o CGP o CGMP o canalizaciones	xsd:string or xsd:integer
tieneIntensidad	Cables	xsd:integer
tieneLíneaDeDistribución	Red de distribución	xsd:string
tieneLongitud	Acometida o DI o LGA o cables	xsd:float
tieneModelo		xsd:string

<b>Propiedades de datos</b>	<b>Dominio</b>	<b>Rango</b>
tieneNaturaleza		xsd:string
tieneNúmero		xsd:int
tieneNombre	Todas las clases	xsd:string
tienePoderDeCorte		xsd:integer
tienePotencia	Red de distribución o ICP	xsd:integer
tienePrecio	Materiales de instalación	xsd:float
tienePrevisiónDeCarga		xsd:integer
tieneProfundidad	Acometida	xsd:float
tieneSección	Cables o canalizaciones o tubos	xsd:integer
tieneSensibilidad	ID	xsd:integer
tieneSistemaDeConexión	Cables	xsd:string
tieneTemperatura	Cables	xsd:integer
tieneTensión	Cables	(xsd:float or xsd:string)
tieneTracciónMáxima Admisible	Cables	xsd:float
tieneUnidad		xsd:string
tieneTipo	Acometida o material de instalación o red de distribución o DGMP o contadores O	xsd:string

A continuación, se aprecia en la imagen cómo quedarían las propiedades de datos implementadas para la ontología de la Instalación Eléctrica en Protégé.





**Figura 12:** Propiedades de dato implementadas en el programa

### 6.3. INDIVIDUOS

Los individuos representan objetos determinados de las clases. Se quiere hacer la ontología de una instalación eléctrica, como el propio nombre del proyecto indica, por lo que es aquí donde implementamos los objetos para crear una instalación eléctrica a través de las clases, y las propiedades definidas con anterioridad en el programa. Para empezar a implementar estos individuos, haremos uso en el programa de la pestaña de “individuals by class” donde se podrá ir añadiendo los individuos por las clases que se han definido. En esta pestaña también se podrá relacionar las distintas clases a través de las propiedades del objeto y definir las características de dicho individuo mediante las propiedades de datos.

En la siguiente imagen se muestra la métrica de la ontología en donde podemos ver el número de axiomas, clases, propiedades e individuos de la que está compuesta la ontología:

Metrics	
Axiom	768
Logical axiom count	530
Declaration axioms count	238
Class count	61
Object property count	39
Data property count	65
Individual count	74
Annotation Property count	1
Class axioms	
SubClassOf	46
EquivalentClasses	0
DisjointClasses	2
GCI count	0
Hidden GCI Count	0

**Figura 13:** Métrica de la ontología

A continuación, se muestran los distintos individuos por clase, implementados para la creación de una instalación eléctrica. Se va a hacer la instalación eléctrica de una casa con cuatro viviendas.

Clases	Individuos	Propiedades de objetos	Propiedades de datos
Acometida	Acometida1	-utiliza Cable	-tieneAltura -tipoDeAcometida
CGP	CGP	-estaEnlazadaCon LGA -esAlimentadaPor Acometida1	
CC	CC	-tieneAlbergados IGM	
CGMP	Cuadro1, Cuadro2, Cuadro3 y Cuadro4	-alojaA DGMP	-descripción
DI	D1, D2, D3, D4	-seAlbergaEn CC -salenDe Cont	
DGMP	DGMP1, DGMP2, DGMP3, DGMP4	protegeA InstInt	
ID	ID1, ID2, ID3, ID4	sonComponentesDe DGMP	-descripción -tieneIntesidad -tieneSensibilidad

IGA	IGA1, IGA2, IGA3, IGA4	sonComponentesDe DGMP	-descripción -tieneIntesidad
PIAs	PIAs1 - PIAs20	sonComponentesDe DGMP	-descripción -tieneIntesidad
Instalaciones interiores	InstInt1, InstInt2, InstInt3, InstInt4	esSuministradoPor DI	
ICP	ICP1, ICP2, ICP3, ICP4	cortaSuministroElectrico De InstInt	
Cables	Cable1, Cable2, Cable3, Cable4	sonUtilizadosPor Red1	-tieneIntensidad -tieneTemperatura -tieneAislamiento -tieneNombre -tipoDeMaterial
Contadores	Cont1, Cont2, Cont3, Cont4	-integranA ICP -seAlberganEn CC	-descripción -tieneDimension
LGA	LGA	-utiliza Cable	
FS	F1 - F7	-seAlberganEn CC -estaConectadoCon Cont	-descripción -tieneIntensidad -tieneTension
Puesta a tierra	PT	aseguranActuacionesDe DGMP	-descripción -tieneIntesidad -tieneColor -tieneSeccion -tieneLongitud -tieneTension
Red de distribución	Red1	distribuyeEnergiaPor Acometida1	-tienePotenciaContra tada -tieneEmpresaSumin istradora
IGM	IGM	desconecta CC	-descripción -tieneIntesidad -tienePotencia -tieneGradoDeProte ccion

Como se puede observar, se ha hecho con el programa, una instalación eléctrica sencilla de una casa con cuadro viviendas en el que se han incluido las relaciones entre las distintas clases y algunas de sus características. Al utilizar el razonador en el programa, este no nos da ninguna inconsistencia, por lo que no hay errores y con esto se deduce que se puede utilizar el programa para crear cualquier instalación eléctrica, pudiendo por supuesto, añadir más propiedades de objetos o datos, si fuera necesario.

A continuación, se aprecia en la imagen cómo quedarían los individuos por clase implementadas para la ontología en Protégé.

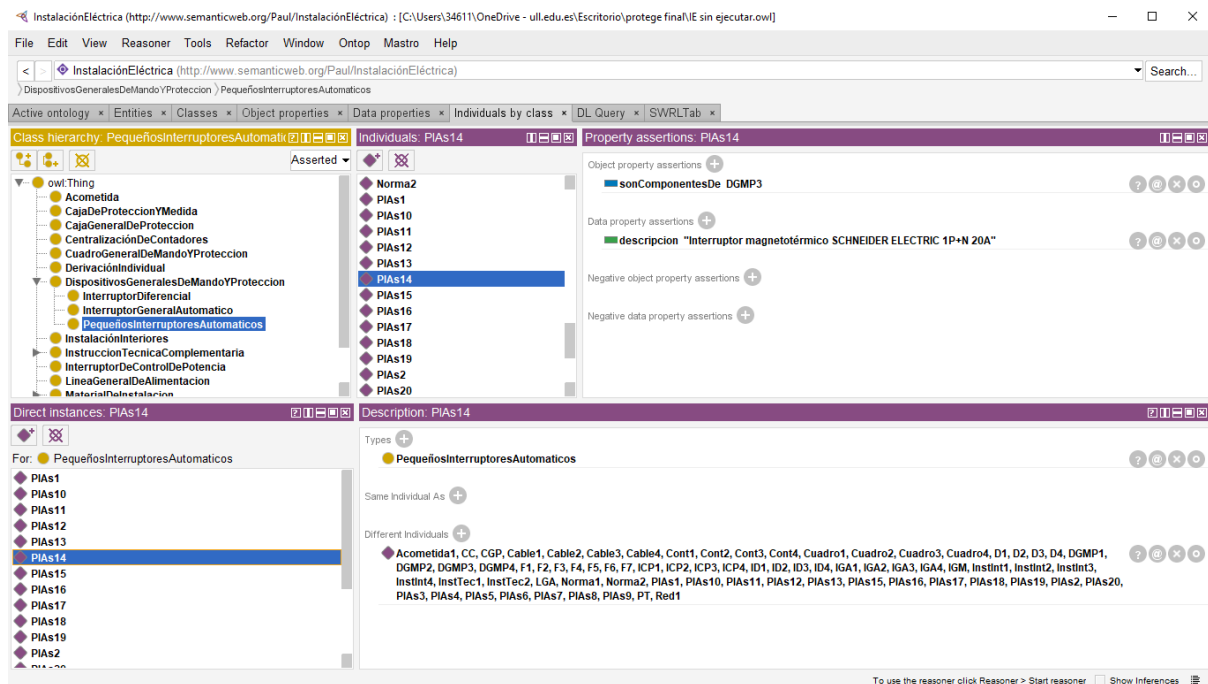


Figura 14: Individuos implementados en el programa

## 6.4. RAZONADORES EN PROTÉGÉ

Una característica fundamental de las ontologías que son descritas usando OWL-DL es que pueden ser procesadas por un razonador. Los razonadores, llamados clasificadores, también pueden comprobar la consistencia.

Un TBox y un ABox son componentes de una base de conocimiento (KB). Un TBox describe conocimiento intencional en la forma de conceptos (clases) y definiciones de relaciones (propiedades). Un ABox trata de un conjunto finito de afirmaciones en cuanto a los individuos al mismo tiempo de utilizar los términos de la ontología. Los ABox solo son una representación incompleta del sobre el mundo. Las clases se organizan en una jerarquía de superclases-subclases, conocido como taxonomía. Estas relaciones pueden ser procesadas por un razonador en OWL-DL. Para la deducción de conocimientos implícitos en la ontología, se debe hacer uso del razonador. Los razonadores pueden ser de dos tipos: razonadores de lógica descriptiva y razonadores de programación lógica.

Los razonadores DL brindan los siguientes servicios de inferencia:

- Validación de la consistencia de una ontología: el razonador comprueba que la ontología no contenga hechos contradictorios.

- Validación del cumplimiento de los conceptos de la ontología: el razonador comprueba si una clase puede tener instancias. En el caso de que no se pueda, se muestra una inconsistencia.
- Clasificación de la ontología: con el objetivo de la construcción de la jerarquía de clases el razonador computará a partir de los axiomas declarados en el TBox las relaciones de subclase entre todos los conceptos declarados explícitamente.
- Posibilita la resolución de consultas durante la recuperación de información basada en ontologías: se puede obtener las subclases, inferir nuevas subclases de un concepto a partir de la jerarquía de clases.
- Permite una mejor precisión en cuanto a las jerarquías de los conceptos: el razonador puede asignar a las clases el individuo o clases correspondiente en la ontología. Esto se produce mediante la jerarquía inferida.

El razonador utilizado en la ontología es el de Pellet. Este razonador es un razonador open source desarrollado por Mindswap. Pellet es un razonador exclusivo para OWL DL e implementado en java.

## **6.5. LENGUAJE DE REGLAS DE LA WEB SEMÁNTICA (SWRL)**

El Lenguaje de reglas de Web Semántica (SWRL) es una combinación de los sublenguajes OWL Lite y OWL DL, lenguajes de la Ontología Web OWL con los sublenguajes Unary / Binary Datalog RuleML del Lenguaje de Marcado de Reglas. La idea es la de aplicar “Datalog RuleML” en nuestra ontología, idea propuesta por W3C en la que se pretende extender el conjunto de axiomas de OWL para reglas semejantes a las de Horn. Por lo que, permite combinar reglas semejantes a las de Horn con una base de conocimientos de OWL.

Esta regla está formada por dos partes, un antecedente (cuerpo) y un consecuente (cabeza). Esto significa que, cuando las condiciones impuestas en el antecedente se cumplen, entonces las impuestas en el consecuente también deben cumplirse.

El antecedente y el consecuente constan de cero o más átomos. Un antecedente sin nada se interpreta como comúnmente verdadero (que quiere decir que está satisfecho por cada interpretación), por lo cual también debe ser verdadero el consecuente; un consecuente sin nada se trata como trivialmente falso (que quiere decir que no está satisfecho por cada interpretación), por lo cual el antecedente también debe ser falso.

La forma de los átomos de estas reglas puede ser de la forma  $X(a)$ ,  $Y(a, b)$ ,  $\text{sameAs}(a, b)$ ,  $\text{differentFrom}(a, b)$ , o  $\text{builtIn}(r, a, \dots)$ , donde  $X$  es una descripción de OWL,  $Y$  es una propiedad de OWL  $r$  es una relación incorporada y  $a$  y  $b$  son variables, individuos OWL o valores de datos OWL.

Una ontología OWL contiene una cadena de axiomas y hechos en sintaxis abstracta. Los axiomas pueden ser de varios tipos, por ejemplo, axiomas de subclase y axiomas de clase. Se propone aportar más sintaxis con axiomas de reglas.

***axioma = regla***

Un axioma de regla está formado por un antecedente (cuerpo) y un consecuente (cabeza), los cuales cada parte está formado por un conjunto de átomos que pueden estar vacío, como se ha comentado con anterioridad. Se puede asignar a un axioma de reglas, una referencia de URI, que sirve para identificar la regla.

En esta sintaxis, una regla tiene la forma:

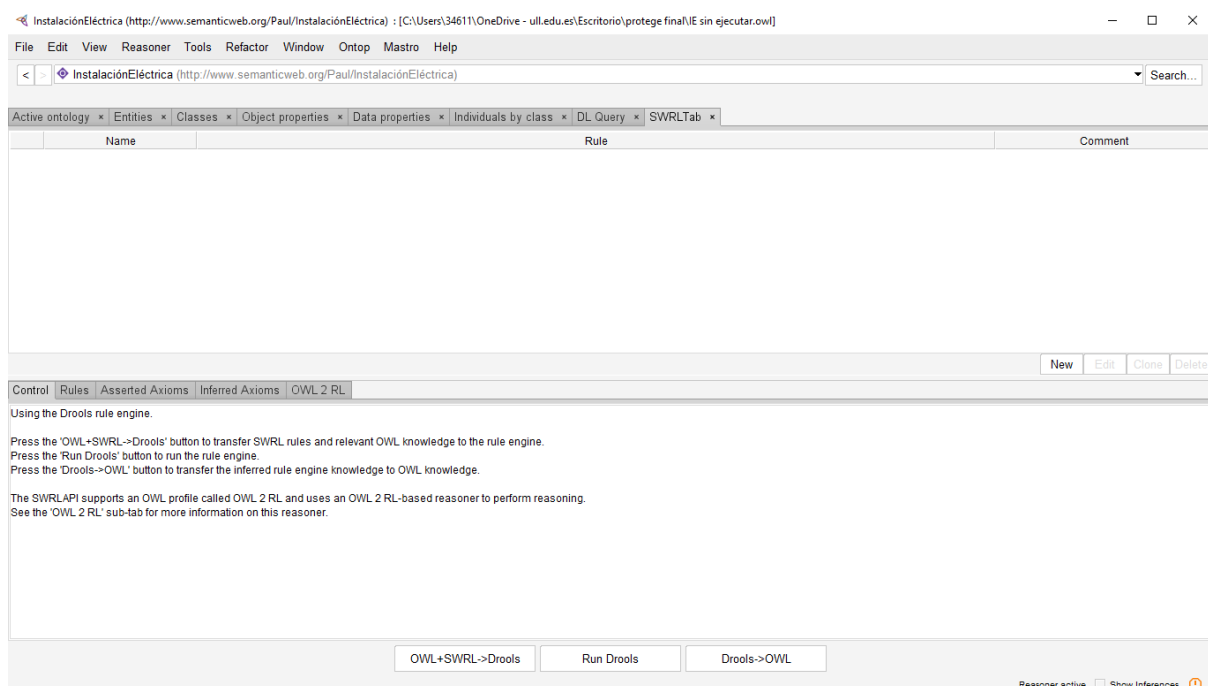
*antecedente*  $\Rightarrow$  *consecuente*

Las variables se indican utilizando la convención estándar de asignarles un signo de interrogación antes de la variable (por ejemplo, ?P), también para la unión de dos interpretaciones se utiliza el acento circunflejo (^) y la coma para separar dos variables. Usando esta sintaxis, una regla que afirma que la composición de las propiedades madre y hermana implica que la propiedad tía se escriba como:

madre (?x, ?y) ^ hermana (?y, ?z)  $\Rightarrow$  tía (?x, ?z)

### 6.5.1. SWRL EN PROTÉGÉ SIN REGLAS

Primero se ejecutará el programa sin reglas para ver todos los axiomas inferidos y para ello, se va a ir a la pestaña SWRL. A continuación, se ve como es la pestaña.



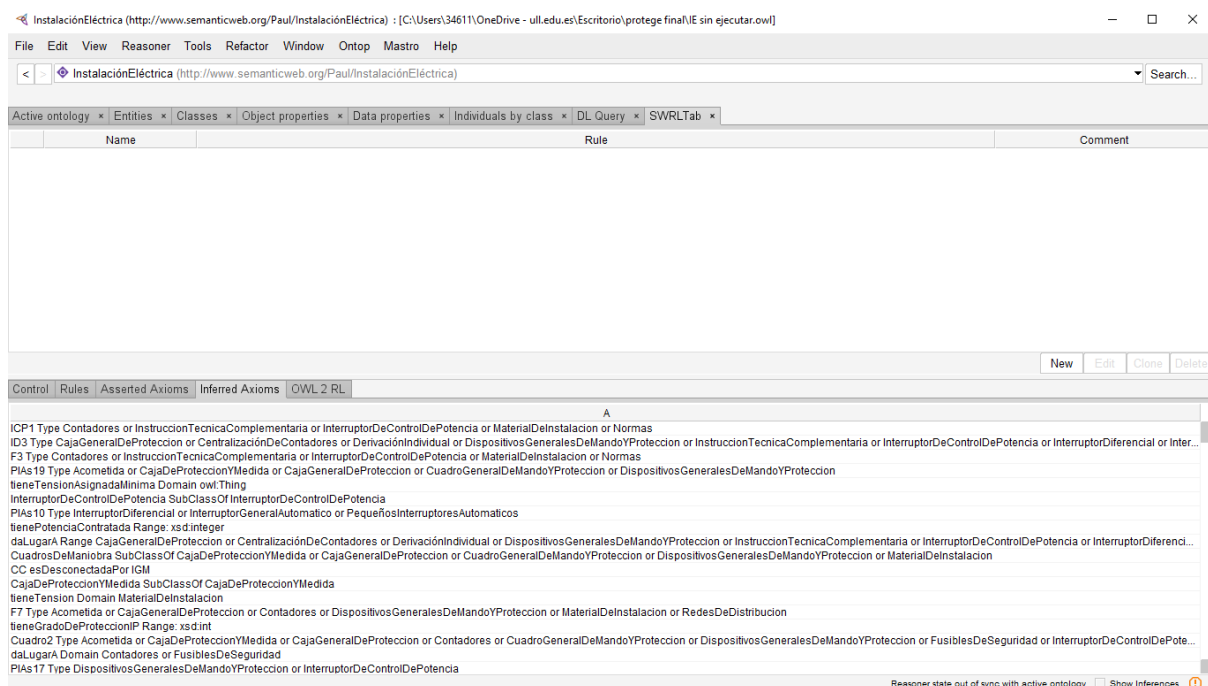
**Figura 15:** Pestaña del lenguaje de regla de la web semántica (SWRL) en Protégé

Se puede observar abajo, que hay tres botones:

- **OWL+SWRL->Drools:** sirve para transferir las reglas SWRL y conocimientos relevantes de OWL al motor de reglas.
- **Run Drools:** sirve para ejecutar el motor de reglas
- **Drools->OWL:** sirve para transferir el conocimiento del motor de reglas inferidas al conocimiento de OWL.

Se debe pulsar estos botones en orden para que aporte nuevos conocimientos a nuestra ontología.

Una vez pulsadas, se puede observar que la pestaña de axiomas inferidos (Inferred Axioms) ya no se encuentra vacía, puesto que se ha transferido nuevos conocimientos a la ontología. Esto se aprecia sobre todo en la pestaña de individuos, en las que se observa que se ha aportado nuevas propiedades de objeto. En las imágenes siguientes, podemos ver la pestaña de los axiomas inferidos y las nuevas propiedades de objetos que fueron aportados por el programa.



**Figura 16:** Axiomas inferidos al ejecutar el programa

En la siguiente imagen se puede observar que aún no se ha ejecutado el programa y que, por ejemplo, el individuo de "Acometida1", solo tiene las propiedades de objeto que le hemos asignado, que sería la de que utiliza los cables 1 y 2.

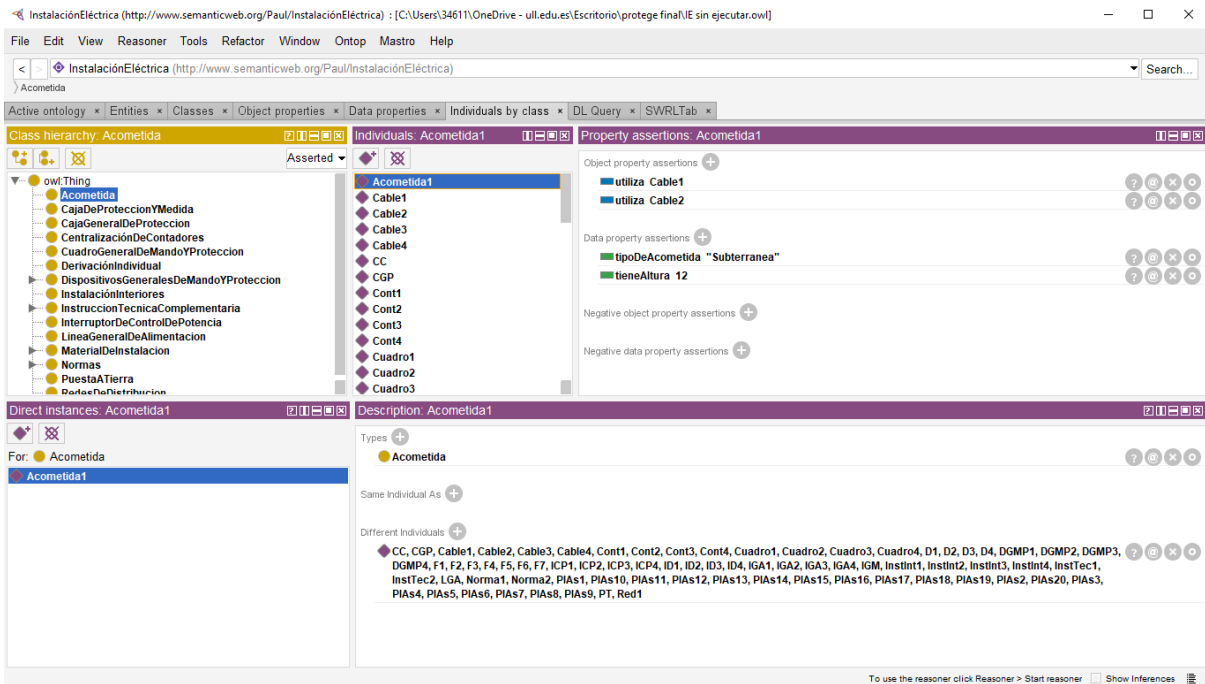


Figura 17: Instancia de Acometida sin ejecutar el programa

Una vez ejecutado el programa, se observa que el programa ha aportado nuevas propiedades de objeto al individuo de “Acometida1”, que sería la de que también utiliza los cables 3 y 4, “obtieneEnergiaDe” “Red1” y que alimenta a la CGP

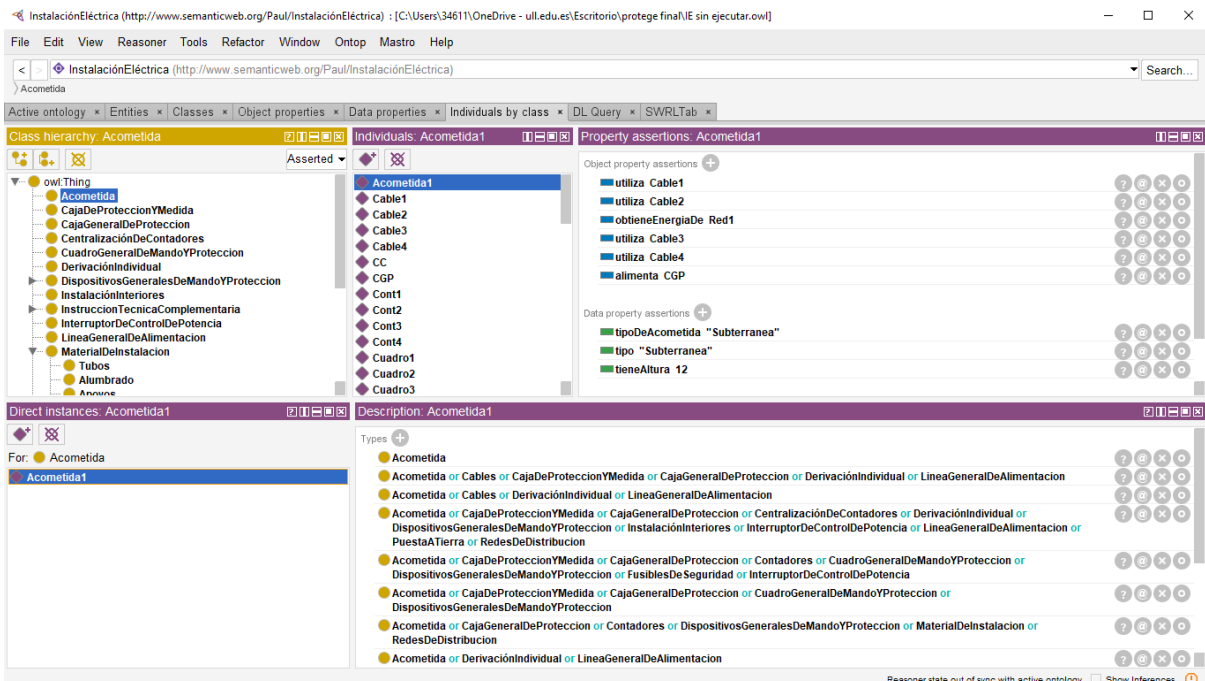


Figura 18: Instancia de Acometida al ejecutar el programa

Lo mismo pasa con la centralización de contadores. Sin ejecutar solo tiene las propiedades que se le ha asignado:



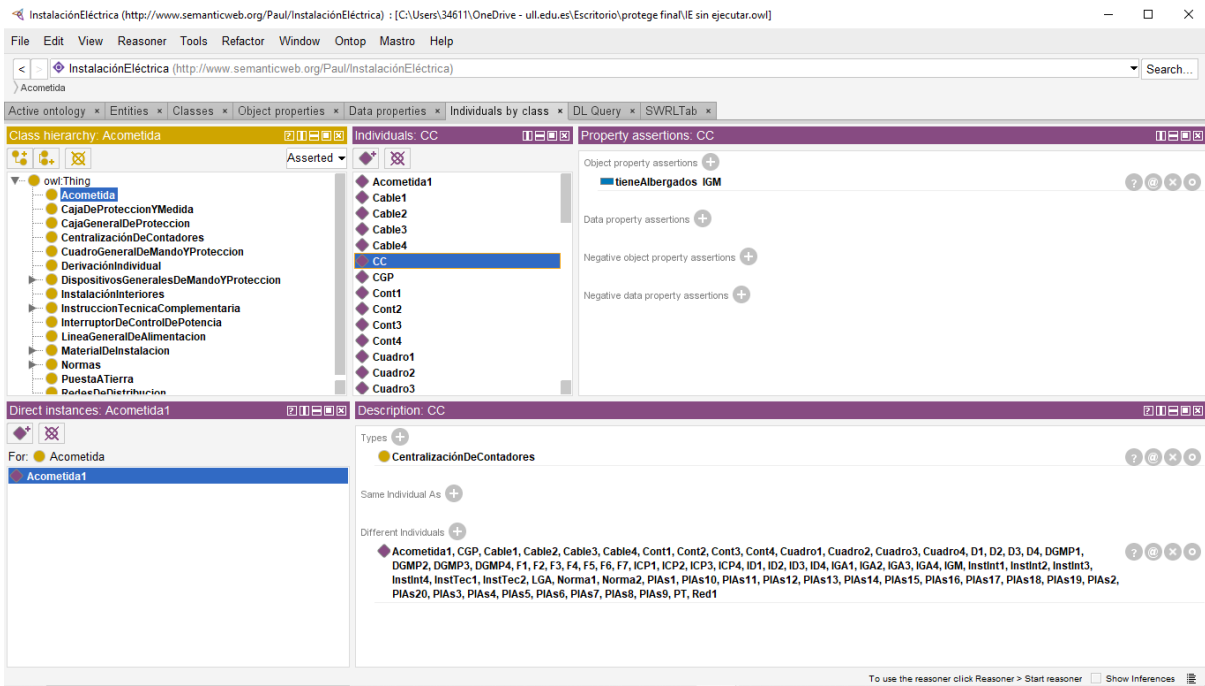


Figura 19: Instancia de centralización de contadores (CC) sin ejecutar el programa

En cambio, al ejecutar el programa, este le aporta nuevos conocimientos que serían en este caso, nuevas propiedades de objeto.

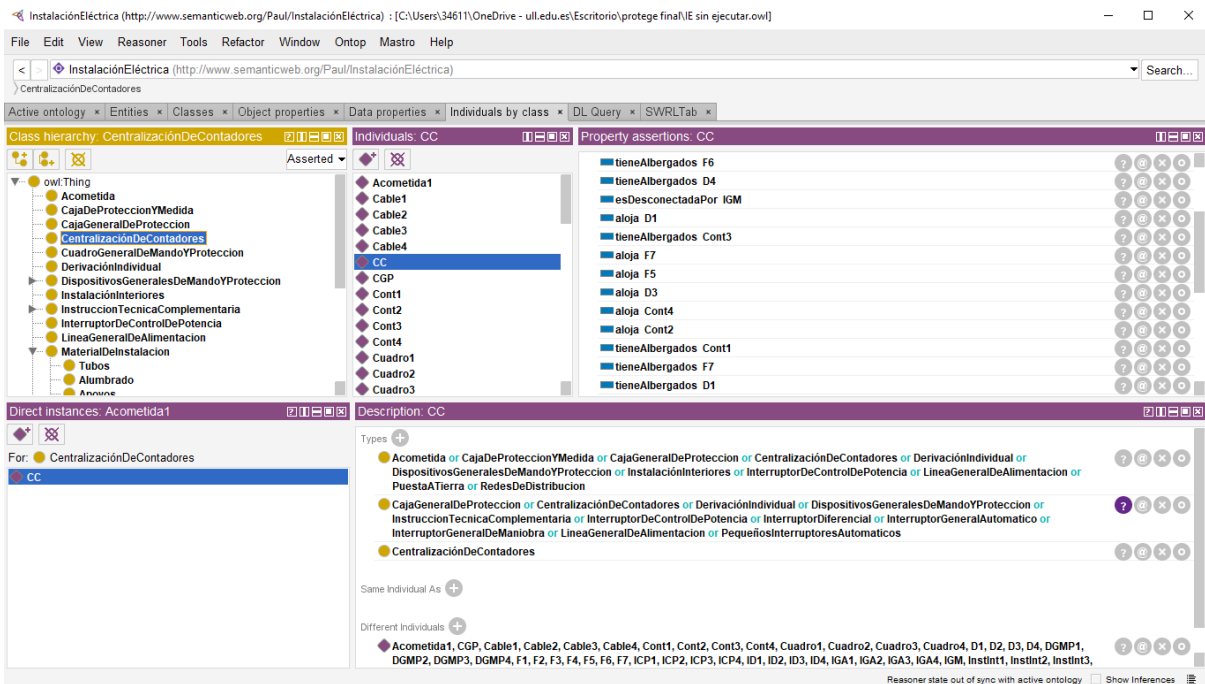
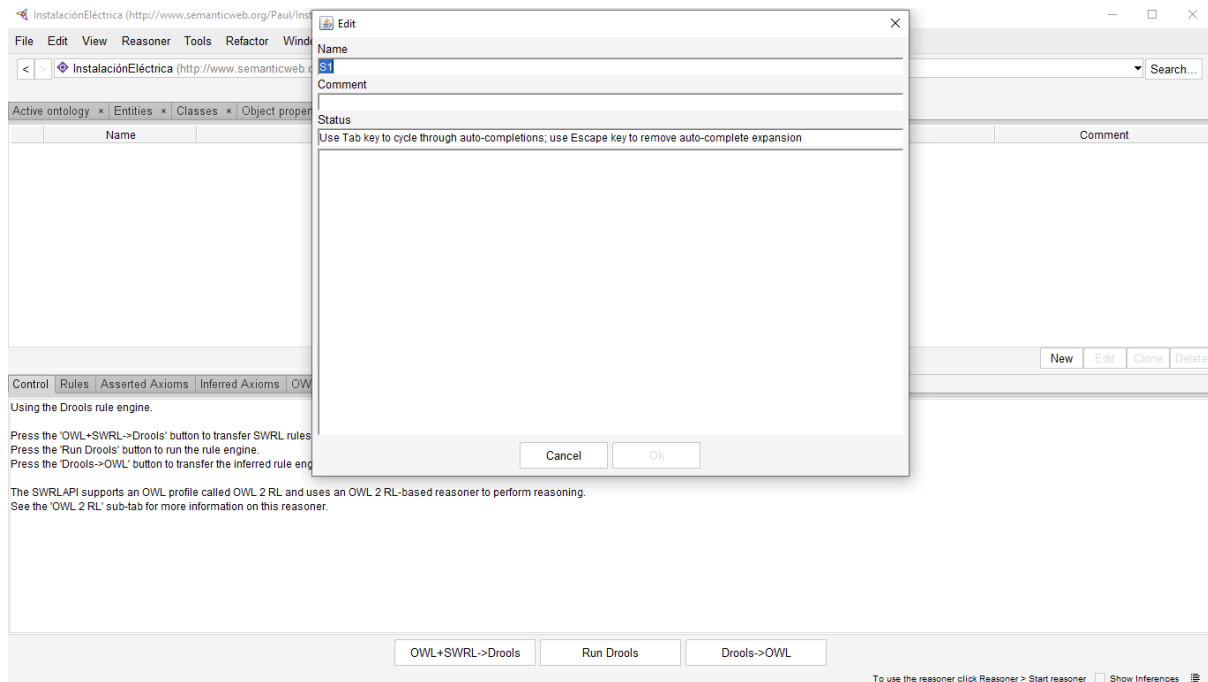


Figura 20: Instancia de centralización de contadores (CC) al ejecutar el programa

## 6.5.2. SWRL EN PROTÉGÉ CON REGLAS

Las reglas sirven para añadir nuevos conocimientos a la ontología. Para añadir una regla, en la pestaña de SWRL se puede observar que hay un botón que pone new (nuevo), al pincharlo, se abrirá una nueva ventana en la que podremos añadir la regla.

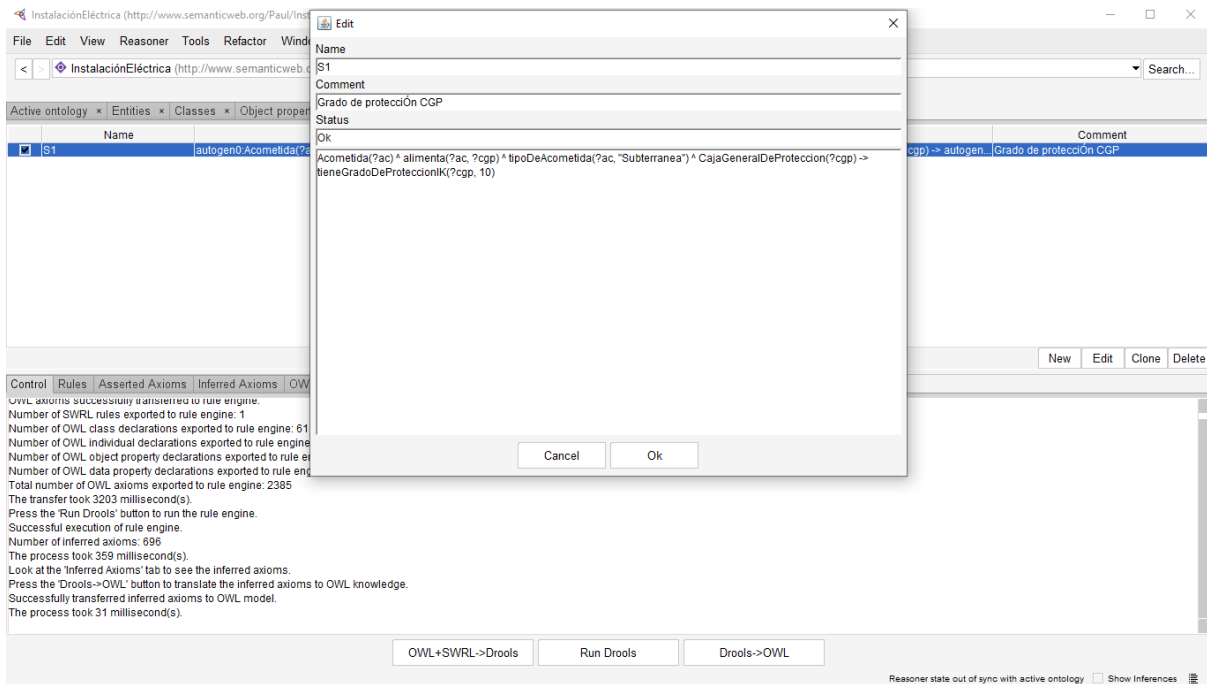


**Figura 21:** Pestaña para introducir las reglas

- La primera regla que se aplicará es para el grado de protección de la caja general de protección. Esta regla se expresa como:

$$\text{Acometida(?ac) \wedge alimenta(?ac, ?cgp) \wedge tipoDeAcometida(?ac, "Subterranea")} \wedge \text{CajaGeneralDeProteccion(?cgp) \rightarrow tieneGradoDeProteccionIK(?cgp, 10)}$$

Esto quiere decir que, la acometida alimenta a la caja general de protección y si la acometida es subterránea entonces como consecuente la CGP debe tener un grado de protección IK de 10 según la norma UNE-EN 50.102 como se indica en el reglamento electrotécnico para baja tensión (REBT), en la ITC-BT-13.



**Figura 22:** Implementación de la regla para el grado de protección de la CGP

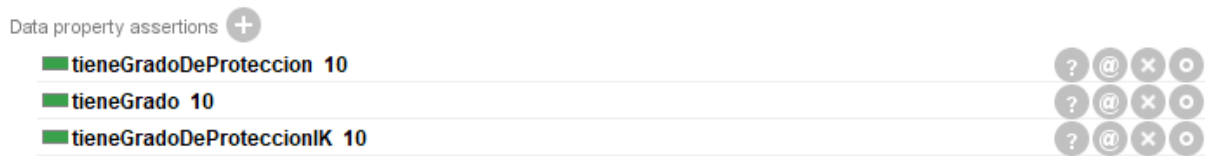
Una vez se ha escrito la regla y que la sintaxis sea correcta (en “Status” se ve un “ok”), el siguiente paso es utilizar el razonador para ver si hay alguna inconsistencia. Una vez comprobado que no hay ninguna inconsistencia o en el caso de que lo haya, solucionado el problema, entonces se ejecuta el programa con los tres botones anteriormente comentados para transferir el conocimiento del motor de reglas inferidas al conocimiento de OWL.

Para comprobar si se ha inferido el nuevo conocimiento a nuestra ontología, se observa si ha habido cambios en el individuo de la Caja general de protección, que se encuentra en la pestaña de individual by class. En la siguiente imagen, se ve que la máquina ha aportado nuevas propiedades de datos al individuo, y esto se debe a que las condiciones impuestas en el antecedente de la regla son verdaderas por lo cual la consecuente también lo es.



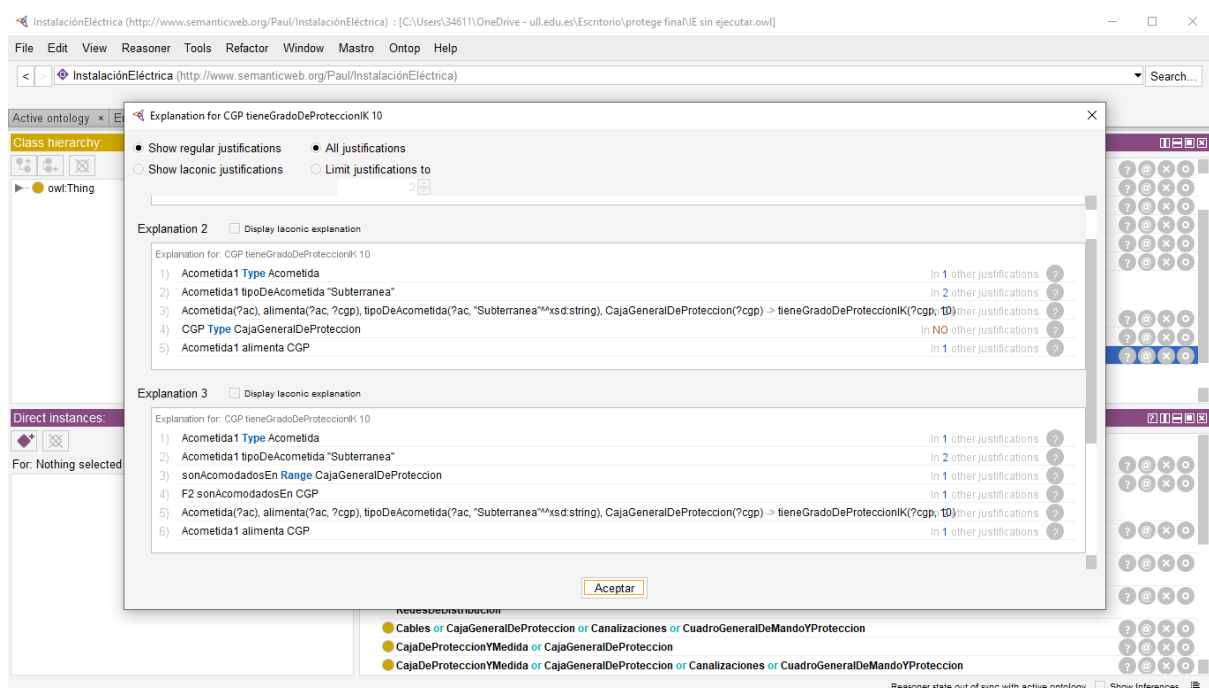
**Figura 23:** Aportación de nuevas propiedades de dato mediante las reglas impuestas

Para comprobar la explicación de por qué la CGP tiene un grado de protección IK de 10, se debe ir a los iconos al lado de “data property assertions” y hacer clic en la que pone interrogación.



**Figura 24:** Propiedad de dato aportada por la regla implementada

Y a continuación, se muestra la explicación de este nuevo conocimiento a nuestra ontología.

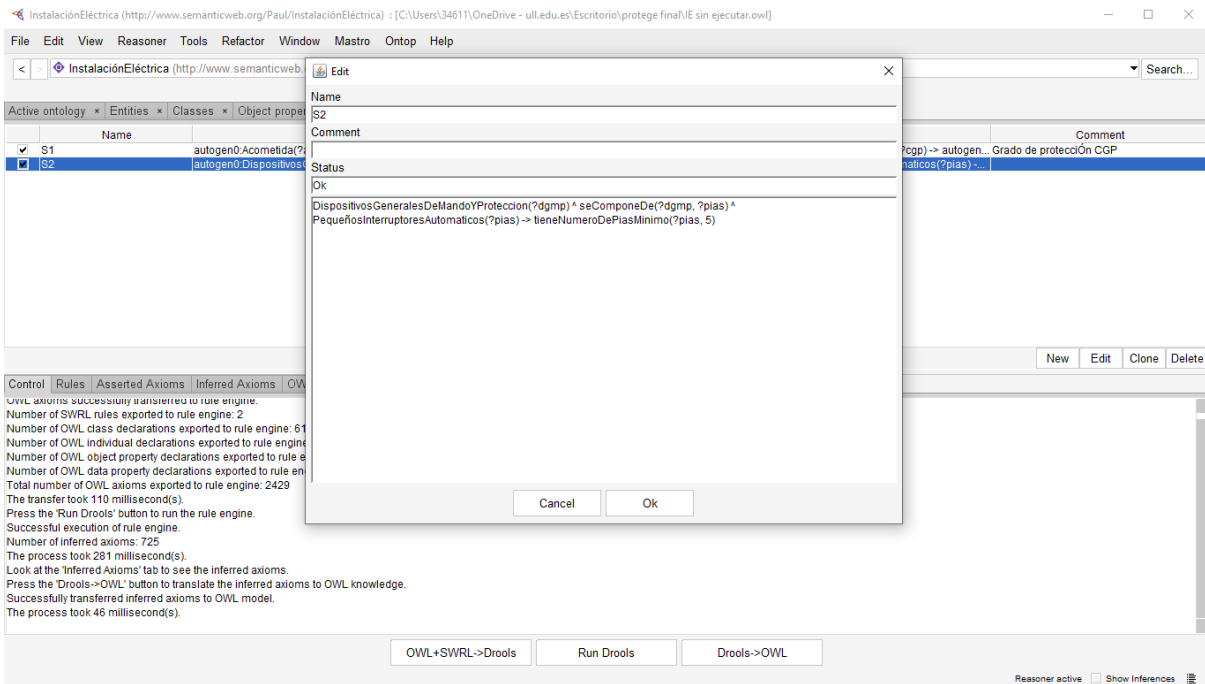


**Figura 25:** Explicación de qué ha inferido en la aportación de la nueva propiedad

Como se puede observar, el programa ha añadido esta nueva propiedad porque en la regla que le hemos impuesto, se han cumplido todas las condiciones del antecedente.

- La siguiente regla que se va a implementar, será para el número mínimo de PIAs (pequeños interruptores automáticos) para cada vivienda. Su regla se expresa como:

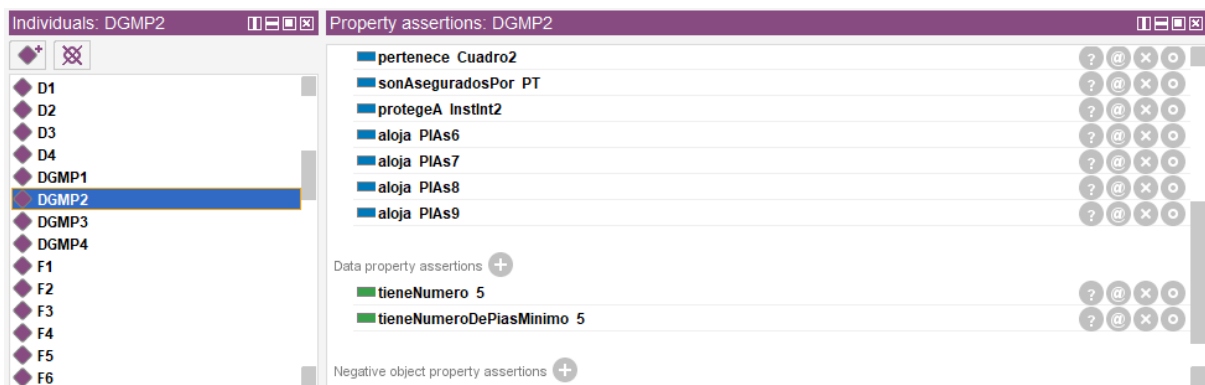
DispositivosGeneralesDeMandoYProteccion(?dgmp) ^ seComponeDe(?dgmp, ?pias) ^ PequeñosInterruptoresAutomaticos(?pias) -> tieneNumeroDePiasMinimo (?pias, 5)



**Figura 26:** Implementación de la regla para el número mínimo de interruptores PIAs de una vivienda

Una vez que todo esté bien y que no haya inconsistencias entonces se ejecuta el programa para que se aporte el nuevo conocimiento que le hemos impuesto a través de la regla.

Comprobamos entonces en la pestaña de “individual by class” si se ha añadido la nueva propiedad de datos que le hemos asignado.



**Figura 27:** Aportación de nuevas propiedades de dato mediante las reglas impuestas

Se observa que efectivamente se ha añadido el nuevo conocimiento impuesto a través de la regla y esto se debe a que se han cumplido todas las condiciones del consecuente de la regla.

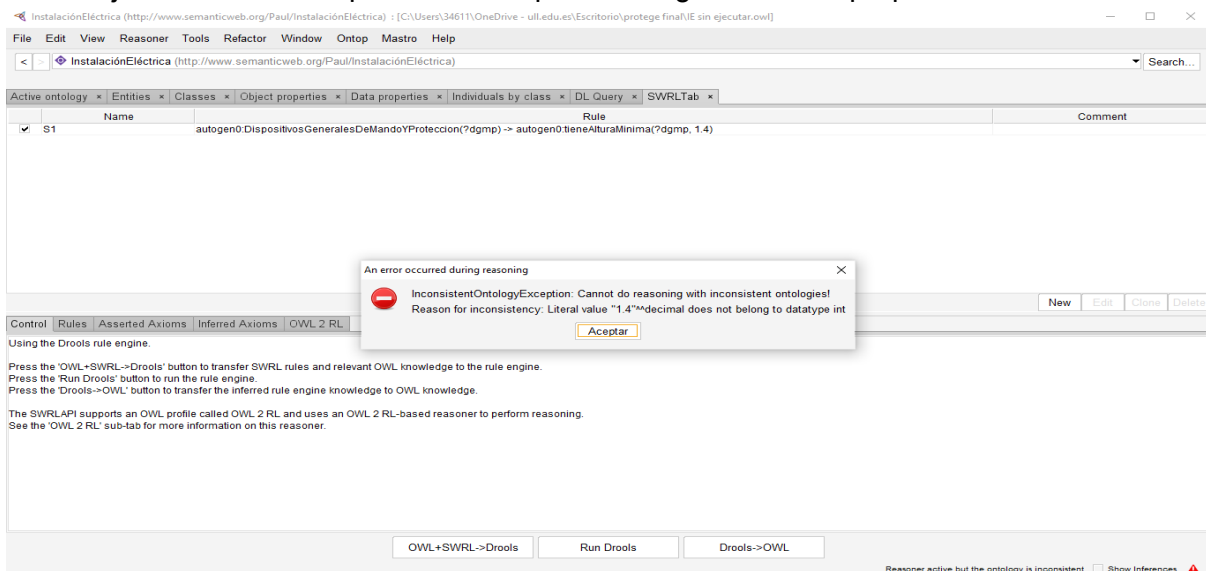
En la siguiente tabla, se muestran todas las reglas que se han implementado en la ontología.

Reglas	Uso
Acometida(?ac) ^ alimenta(?ac, ?cgp) ^ tipoDeAcometida(?ac, "Subterranea") ^ CajaGeneralDeProteccion(?cgp) -> tipoDeInstalacion(?cgp, " En un nicho de la pared")	Tipo de instalación CGP
Acometida(?ac) ^ alimenta(?ac, ?cgp) ^ tipoDeAcometida(?ac, "Subterranea") ^ CajaGeneralDeProteccion(?cgp) -> tieneGradoDeProteccionIK(?cgp, 10)	Grado de protección IK CGP
Cables(?c) ^ Acometida(?ac) ^ utiliza(?ac, ?c) -> tipoDeConductor(?c, "Aislado")	Tipo de conductores
Acometida(?ac) -> tieneComposicion(?ac, "3 cables fases y 1 neutro")	Composición de la acometida
Acometida(?ac) ^ alimenta(?ac, ?cgp) ^ tipoDeAcometida(?ac, "Subterranea") ^ autogen0:CajaGeneralDeProteccion(?cgp) -> tieneAlturaMinima(?cgp, 0.3)	Altura mínima de la CGP
Cables(?c) ^ tipoDeConductor(?c, "Aislado") -> tieneTensionAsignadaMinima (?c, "0.6/1Kv")	Tensión asignada mínima de los conductores
DispositivoGeneralDeMandoYProteccion(?dgmp) -> tieneEmplazamiento(?dgmp, "Al lado de la puerta de entrada")	Emplazamiento del DGMP
DispositivoGeneralDeMandoYProteccion(?dgmp) -> tieneAlturaMaxima(?dgmp, 2)	Altura máxima del DGMP
DispositivoGeneralDeMandoYProteccion(?dgmp) -> tieneAlturaMinima(?dgmp, 1.4)	Altura mínima del DGMP
DispositivoGeneralDeMandoYProteccion(?dgmp) ^ seComponeDe(?dgmp, ?pias) ^ PequeñosIntERRUPTORESAutomaticos(?pias) -> tieneNumeroDePiasMinimo (?pias, 5)	Número mínimo de PIAs para cada vivienda
CuadroGeneralDeMandoYProteccion(?cgmp) -> tieneGradoDeProteccionIPMinimo (?cgmp, 30)	Grado de protección IP mínimo del CGMP
CuadroGeneralDeMandoYProteccion(?cgmp) -> tieneGradoDeProteccionIKMinimo (?cgmp, 07)	Grado de protección IK mínimo del CGMP
Cables(?c) ^ tipoDeConductores(?c, "Aislado") ^ tipoDeMaterial(?c, "Cobre") ->	Sección mínima de los conductores de cobre

seccionMinima(?c, 6)	
Cables(?c) ^ tipoDeConductores(?c, "Aislado") ^ tipoDeMaterial(?c, "Aluminio") -> seccionMinima(?c, 16)	Sección mínima de los conductores de aluminio
PuestaATierra (?pt) -> tieneColor(?pt, "Amarillo y Verde")	Color del cable de puesta a tierra

## 7. PROBLEMAS A LO LARGO DE LA EJECUCIÓN DEL PROGRAMA Y SUS SOLUCIONES

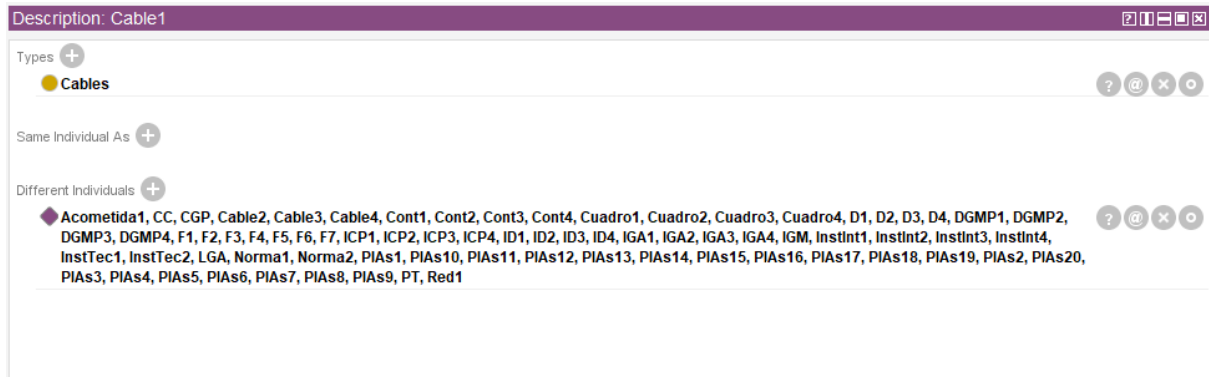
- Las inconsistencias al utilizar el razonador: Existen diferentes formas en las que puede saltar el cuadro de inconsistencia al empezar el razonador como, por ejemplo, cuando se añade un individuo a dos clases disjuntas, o cuando una propiedad es funcional y se añade dos o más veces esa propiedad para un solo individuo. Pero en este caso, a la hora de ejecutar el programa, las inconsistencias eran por los literales, es decir, se tenía que vigilar los "data type" que eran asignados al rango de las propiedades de datos porque dependiendo de valor que se le atribuye, este puede ser o uno u otro tipo de dato. Para la solución de este problema solo hay que fijarse bien en los tipos de datos que son asignados a las propiedades de datos.



**Figura 28:** Inconsistencia producido por el programa

- Problemas a la hora de asignar las clases y subclases: Siempre se tiene que, tener en cuenta que, para la asignación de las subclases, se debe hacer la pregunta: ¿Es X un Y? donde "X" hace referencia a las subclases e "Y" a las clases. Si esta pregunta es afirmativa, entonces se puede asignar esa subclase.
- Instancias asignadas a clases incorrectas: Este problema se produce al no especificarle al programa que todos los individuos son distintos, por tanto, este

asume que dos individuos son iguales si una propiedad se les asignan a ambos. Hay que saber que, en la web semántica, se trabaja con una asunción de mundo abierto, es decir, que si no se especifica algo no significa que sea falso. Por lo que, si no se indica que son diferentes, no se puede presuponer. Y puede ocurrir que mediante un proceso de razonamiento se puede llegar a la conclusión de que dos individuos son iguales o pueden pertenecer a la misma clase. Para solucionar este problema solo hace falta especificarle al programa que todos los individuos son distintos y esto se hace en la pestaña de “Individual by class”, en la descripción, y añadiendo esto en la casilla de “Different Individuals”.



**Figura 29:** Especificación de la diferencia entre todos los individuos



## 8. PRESUPUESTO DEL PROYECTO

Para la realización de este proyecto, solo ha hecho falta un ordenador y horas de trabajo.

- El ordenador utilizado es el HP Laptop 15-db0xxx, con las siguientes características:

- Nombre del dispositivo: LAPTOP-DS9Q42N8
- Procesador: AMD A6-9225 RADEON R4, 5 COMPUTE CORES 2C+3G 2.60 GHz
- RAM instalada 8,00 GB (7,46 GB usable)
- Tipo de sistema: Sistema operativo de 64 bits, procesador basado en x64

- Las horas de trabajo fueron aproximadamente de 175 horas. Comenzando desde el lunes 21 de marzo hasta el viernes 28 de mayo, con un total de 50 días (una media de 3,5 horas diarias) trabajados excluyendo los fines de semana.

<b>PRESUPUESTO</b>
--------------------

ITEM	CANTIDAD	COSTO	VALOR
Horas de trabajo	175h	12€	2100€
Ordenador	1 unidad	400€	400€
Ratón	1 unidad	10€	10€
Total			2510€

## 9. CONCLUSIONES

From my point of view and with the research of information that comes from different sources, I believe that the semantic web is a very long-term project, because although it is a very good idea, its development is being carried out slowly due to the lack of funds and Interest that are required for how extensive it is. It would take a lot of qualified workers and a lot of time to be able to introduce this extension to the web, since there is an overwhelming amount of information.

The semantic web is necessary to facilitate more precisely the search and retrieval of information, not only for those who are quite familiar with the web but especially for inexperienced people (such as the elderly or children). Who has not searched for a topic by asking a question in the search engine? The search engine has no way of thinking or responding with the correct answer, but rather according to the words we enter, it searches its entire database, which may or may not be the information we are looking for. With the semantic web, we will provide computers with the necessary resources to be more concise in relation to our demand for information.

Regarding the construction of the ontology of an electrical installation, this has been quite simple since it has been based on the low voltage electrotechnical regulation (REBT) where the components of an installation and the relationships that exist between them can be consulted. A general ontology of an electrical installation has been made, and the classes may have been broken down into more subclasses and therefore more properties could have been included, extending the ontology further, but as I have said previously, it would take much more time.

It has been possible to adapt the ontology for any electrical installation, therefore it could be said that the ontology is fine, although of course it could be extended much more with the contribution of more specifications.

# REFERENCIAS

- [1] (marzo 19,). *¿Qué son los metadatos y cuál es su utilidad?*. Available: <https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/que-son-los-metadatos-y-cual-es-su-utilidad#:~:text=Es%20com%C3%BAn%20describir%20el%20t%C3%A9rmino,la%20informaci%C3%B3n%20de%20los%20mismos.>
- [2] (Jul 12,). *XML: ¿Qué es y para qué sirve este lenguaje de marcado?*. Available: <https://rockcontent.com/es/blog/que-es-xml/#:~:text=XML%20es%20el%20acr%C3%B3nimo%20de,para%20la%20codificaci%C3%B3n%20de%20documentos.&text=Un%20archivo%20XML%20se%20divide%20en%20dos%20partes%3A%20prolog%20y%20body.>
- [3] (Julio 29,). *HIPERTEXTO: EL NUEVO CONCEPTO DE DOCUMENTO EN LA CULTURA DE LA IMAGEN*. Available: <http://www.hipertexto.info/>.
- [4] (Junio). *Construcción de una ontología OWL con protégé 4*. Available: <https://es.slideshare.net/tanianar/construccin-de-una-ontologa-owl-con-protg-4>.
- [5] Anonymous "REGLAMENTO ELECTROTÉCNICO PARA BAJA TENSIÓN E INSTRUCCIONES TÉCNICAS COMPLEMENTARIAS (ITC) BT 01 A BT 51," .
- [6] , E T I E, "Instalación eléctrica en una nave industrial destinada a la fabricación de muebles de oficina," .
- [7] A. Héctor and Flórez Fernández, "Construcción de ontologías OWL," .
- [8] R. Romero Llop, "Especificación OWL de una ontología para teleeducación en WEB semántica," 2011. Available: [https://explore.openaire.eu/search/publication?articleId=dedup\\_wf\\_001::73809088c219782d0557a1a1014c1f97](https://explore.openaire.eu/search/publication?articleId=dedup_wf_001::73809088c219782d0557a1a1014c1f97). DOI: 10.4995/Thesis/10251/1828.
- [9] A. Casali and H. Gonzalez, "Introducción a Protégé 5.2.0 Transparencias base de Flavio E. Spetale (IIA-LCC) y T. Rodríguez y J. Aguilar "A Practical Guide To Building OWL Ontologies Using Protégé 4"," .
- [10] M. Horridge *et al*, "A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools Edition 1.3," 2011.
- [11] Anonymous "SWRL: A Semantic Web Rule Language Combining OWL and RuleML W3C Member Submission 21 May 2004," .
- [12] M. O'Connor, "SWRLTab: A Development Environment for working with SWRL Rules In Protégé-OWL," .
- [13] R. Pedraza-Jiménez, L. Codina and C. Rovira, "Web semántica y ontologías en el procesamiento de la información documental," *El Profesional De La Informacion*, vol. 16, (6), pp. 569-578, 2007. Available: [https://explore.openaire.eu/search/publication?articleId=dedup\\_wf\\_001::871d690618036892d96704d7617a9662](https://explore.openaire.eu/search/publication?articleId=dedup_wf_001::871d690618036892d96704d7617a9662). DOI: 10.3145/epi.2007.nov.04.
- [14] J. Cáceres Tello, ""La Web Semántica y el lenguaje RDF"," 2014.

[15] Anonymous "TFC-WEB SEMANTICA MEMORIA FINAL ALUMNO SINESIO DAVID CARVAJAL TABASCO INGENIERIA TECNICA INFORMÁTICA DE GESTIÓN CONSULTOR SINUHE ARROYO GÓMEZ," .

[16] J. A. Senso and J. A. Senso, "El concepto de metadato. Algo más que descripción de El concepto de metadato. Algo más que descripción de recursos electrónicos recursos electrónicos Antonio de la Rosa Piñero Antonio de la Rosa Piñero," .

[17] U. Entorno *et al*, "UNIVERSIDAD DE MURCIA Departamento de Ingeniería de la Información y las Comunicaciones TESIS DOCTORAL," .

[18] F. F. Martínez Arellano and M. A. Amaya Ramírez, "El papel de los metadatos en la Web Semántica," *Biblioteca Universitaria (Mexico City, Mexico)*, vol. 20, (1), pp. 3-10, 2017. . DOI: 10.22201/dgb.0187750xp.2017.1.171.

[19] C. Bravo Santos y M. Á. Redondo Duque, *Sistemas Interactivos Y Colaborativos En La Web*. (Universidad de Castilla-La Mancha ed.) Cuenca: 2005.

[20] Y. Hidalgo Delgado and R. Rodríguez Puente, "La web semántica: una breve revisión," *Revista Cubana De Ciencias Informáticas*, vol. 7, (1), pp. 76-85, 2013.

Available:

[http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S2227-18992013000100009&lng=en&tlng=en](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2227-18992013000100009&lng=en&tlng=en).

[21] Juan Antonio Pastor Sánchez, *Tecnología De La Web Semántica*. El Ciervo 96, S.A, 2011.

[22] R. G. Ruiz, "TFC: XML y Web Semántica Estudio del Impacto de las aplicaciones comerciales basadas en tecnologías de Web Semántica," 2007.

[23] Tim Berners-Lee. *Semantic Web -XML2000. Architecture* Available:

<http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide11-0.html>

[24] M. Horridge *et al*, "A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools Edition 1.3," 2011.

[25] Stanford center for biomedical informatics. Stanford University. Available:

<https://protege.stanford.edu/>

La descarga de la ontología de la instalación eléctrica se concede a través del siguiente enlace:

<https://drive.google.com/file/d/11RNgcgqVMkiMPrwqO0wW52KiZUdXtmDz/view?usp=sharing>