

**UNIVERSIDAD PRIVADA ANTENOR ORREGO  
FACULTAD DE INGENIERÍA  
ESCUELA PROFESIONAL DE INGENIERÍA DE  
COMPUTACIÓN Y SISTEMAS**



---

**“ONTOLOGÍA DE PATRÓN DE COMPORTAMIENTO COMO CONTROLADOR  
WEB DE DISPOSITIVOS EN UN DEPARTAMENTO DOMOTIZADO DE LA  
CONSTRUCTORA PROARCO S.A.C.”**

---

**TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE  
INGENIERO DE COMPUTACIÓN Y SISTEMAS  
LÍNEA DE INVESTIGACIÓN: WEB SEMÁNTICA**

**AUTOR:** Br. Cacho Aniceto, Luz Stefany  
Br. Torres Sáenz, Joaquín Osmín Aarón

**ASESOR:** Dr. Luis Vladimir Urrelo Huiman

**TRUJILLO - PERÚ**

**2016**

## **ACREDITACIONES**

**Tesis: “ONTOLOGÍA DE PATRÓN DE COMPORTAMIENTO COMO CONTROLADOR WEB DE DISPOSITIVOS EN UN DEPARAMENTO DOMOTIZADO DE LA CONSTRUCTORA PROARCO S.A.C.”**

Por: Br. Cacho Aniceto, Luz Stefany

Br. Torres Sáenz, Joaquín Osmín Aarón

APROBADO POR:

---

Ing. Elmer González Herrena  
PRESIDENTE  
N° CIP 24721

---

Ing. Jorge Piminchumo Quispe  
SECRETARIO  
N° CIP 137153

---

Ing. Agustín Ullón Ramírez  
VOCAL  
N° CIP 137602

---

Ing. Luis Vladimir Urrelo Huiman  
ASESOR  
N° CIP 88212

## **PRESENTACION**

**Señores Miembros del Jurado:**

Dando cumplimiento y conforme a las normas establecidas en el Reglamento de Grados y Títulos y Reglamento de la Facultad de Ingeniería de la Universidad Privada Antenor Orrego, para obtener el título profesional de Ingeniero Computación y Sistemas, se pone a vuestra consideración el Informe del Trabajo de Investigación Titulado “**ONTOLOGÍA DE PATRÓN DE COMPORTAMIENTO COMO CONTROLADOR WEB DE DISPOSITIVOS EN UN DEPARAMENTO DOMOTIZADO DE LA CONSTRUCTORA PROARCO S.A.C.**”, con la convicción de alcanzar una justa evaluación y dictamen, excusándonos de antemano de los posibles errores involuntarios cometidos en el desarrollo del mismo.

Atentamente,

Trujillo, abril del 2016

Br. Cacho Aniceto, Luz Stefany

Br. Torres Sáenz, Joaquín Osmín Aarón

## **DEDICATORIA**

A mi familia, que me dieron el empujón a la vida y me enseñaron a ser este individuo.

Mis amigos de la carrera, mi segunda familia, por el apoyo hombro a hombro durante el aprendizaje.

A todos aquellos que encuentran en la tecnología e investigación los instrumentos para innovar todos los ámbitos del pensamiento humano.

Br. Torres Sáenz, Joaquín Osmín Aarón

A Dios por haberme permitido llegar hasta este punto, haberme dado salud y fortaleza para lograr mis objetivos.

A mi Papi César y mami Luz, que me apoyaron en mis decisiones y motivaron a cumplirlas, sobre todo por creer siempre en mí que lograría cada meta establecida.

A mi hermano Anghelo o como le digo mi Bichito que es mi motor para ser mejor cada día.

Br. Cacho Aniceto, Luz Stefany

## **AGRADECIMIENTO**

Expresamos nuestro agradecimiento a nuestros padres por apoyarnos en todo momento, por los valores que nos han inculcado, y por habernos dado la oportunidad de tener una excelente educación en el transcurso de nuestras vidas.

A nuestro agradable asesor Ing. Luis Vladimir Urrelo Huiman por sus conocimientos, sus orientaciones, su paciencia y su motivación, las cuales han sido fundamentales para nuestra formación como investigadores. Inculcando en nosotras seriedad y responsabilidad durante el periodo de tiempo invertido el desarrollo de esta tesis.

A todos los Ingenieros especialistas y a nuestra alma mater UPAO, porque durante los cinco años forjaron en nosotros los saberes supremos de carácter científico y humanístico, transformándonos en un mejor y autentico ser humano para la vida.

Br. Cacho Aniceto, Luz Stefany

Br. Torres Sáenz, Joaquín Osmín Aarón

## RESUMEN

### **“ONTOLOGÍA DE PATRÓN DE COMPORTAMIENTO COMO CONTROLADOR WEB DE DISPOSITIVOS EN UN DEPARTAMENTO DOMOTIZADO DE LA CONSTRUCTORA PROARCO S.A.C.”**

**Por el Br. Cacho Aniceto, Luz Stefany**

**Br. Torres Sáenz, Joaquín Osmín Aarón**

La Constructora PROARCO S.A.C, es una empresa dedicada a la construcción y venta de departamentos, cuenta con el proyecto de Residencial Los Jazmines, en donde se implementará un departamento domótico. La empresa requiere que este departamento domótico tenga un sistema más inteligente que a diferencia de los existentes, se anticipe a las preferencias de los usuarios del departamento, sin incurrir en un alto costo. En tal sentido ¿Qué componente de Tecnologías de Información permitirá anticiparse a las preferencias de los usuarios en un departamento domotizado del proyecto Residencial Los Jazmines de la empresa PROARCO S.A.C., la presente investigación propone el desarrollo de una Ontología de patrón de comportamiento como controlador Web que permitirá anticiparse a las preferencias de los usuarios en el departamento domotizado 301 del proyecto Residencial Los Jazmines de la empresa PROARCO S.A.C., para ello se desarrolló la ontología de patrón de comportamiento como controlador Web, identificándose las características que tiene un departamento domotizado, mediante la revisión bibliográfica, se realizó un análisis de los términos y taxonomía de una Ontología con dispositivos de un departamento domotizado y patrón de comportamiento de personas habitantes en dicho departamento según la metodología Methontology y reutilizando conceptos de las ontologías DogOnt y DogPower, definiéndose relaciones binarias de una Ontología, sus atributos, axiomas y reglas de inferencia e implementándose la Ontología desarrollada anteriormente utilizando el software Protégé y el razonador Pellet; concluyendo que el uso de la Ontología desarrollada logra anticiparse a las preferencias de los usuarios habitantes en el departamento domotizado utilizando TI mediante cuatro reglas de inferencia.

Palabras clave: Ontología, Domótica.

## ABSTRACT

### **“ONTOLOGÍA DE PATRÓN DE COMPORTAMIENTO COMO CONTROLADOR WEB DE DISPOSITIVOS EN UN DEPARTAMENTO DOMOTIZADO DE LA CONSTRUCTORA PROARCO S.A.C.”**

**Por el Br. Cacho Aniceto, Luz Stefany**

**Br. Torres Sáenz, Joaquín Osmín Aarón**

The building company PROARCO S.A.C, is dedicated to the construction and sale of apartments they have a residential project called “Los Jazmines”, where a home automation department will be implemented. The company requires that the automation department has a smarter system that unlike the existing ones, anticipating to the user’s preferences. In this regard, which information technology component will allow to anticipate to the user’s preferences in a domotic department from the Residential Project “Los Jazmines” from the building company PROARCO S.A.C.? To this end, we propose the development of an Ontology behavior pattern as a Web controller, the one that will allow the anticipation to the users' preferences in the domotic department 301 located in the Residential Project “Los Jazmines” from the building company PROARCO SAC, for this an ontology pattern has been developed, an analysis of the terms and taxonomy of Ontology with devices from a domotic department and behavioral pattern of people in that department according to Methontology methodology and reusing concepts of DogOnt and DogPower ontologies, definition of binary relations, attribute definition, axioms and rules of inference of a ontology according to the Methontology methodology were performed and it was developed using the Protégé software and Pellet reasoner; concluding that the use of the developed Ontology achieves the anticipation to the preferences of people in the domotic department through four inference rules.

Keywords: Ontology, Home Automation.

# INDICE DE CONTENIDO

ACREDITACIONES .....	ii
PRESENTACION.....	iii
DEDICATORIA .....	iv
AGRADECIMIENTO .....	v
RESUMEN .....	vi
ABSTRACT .....	vii
CAPITULO I: INTRODUCCION.....	13
1.1. Planteamiento del problema .....	13
1.2. Delimitación del problema:.....	16
1.3. Características problemáticas: .....	16
1.4. Formulación del problema: .....	17
1.5. Formulación de la Hipótesis:.....	17
1.6. Objetivos del estudio: .....	17
1.6.1. Objetivo General .....	17
1.6.2. Objetivos Específicos.....	18
1.7. Justificación del Estudio: .....	18
1.8. Limitaciones del Estudio:.....	19
CAPITULO II: MARCO TEÓRICO .....	19
CAPITULO IV: RESULTADOS.....	33
CAPITULO V: DISCUSION DE RESULTADOS.....	94
CAPITULO VI: CONCLUSIONES.....	112
CAPITULO VIII: RECOMENDACIONES.....	113
<b>REFERENCIAS BIBLIOGRÁFICAS</b> .....	114
ANEXOS .....	118
ANEXO 01: Entrevista realizada al Consultor de TI de la Constructora.....	119
ANEXO 02: Planos de un nivel tipo del edificio multifamiliar Jardines del Golf de PROARCO S.A.C. ....	120
ANEXO 03: Planos en 3D de un nivel tipo del edificio multifamiliar Jardines del Golf de PROARCO S.A.C. ....	121

## TABLA DE ILUSTRACIONES

Ilustración 1: Modelado Conceptual de una Ontología .....	23
Ilustración 2: La Ontología .....	25
Ilustración 3: DogOnt 2.5. ....	27
Ilustración 4: Ontologías DogOnt y DogPower.....	28
Ilustración 5: Tareas de conceptualización en Methontology .....	34
Ilustración 6: Clase BuildingEnvironment .....	35
Ilustración 7: Clase GeoLocation .....	35
Ilustración 8: Clase Controllable.....	36
Ilustración 9: Clase Functionality .....	36
Ilustración 10: Clase Command.....	37
Ilustración 11: Clase State y StateValue.....	38
Ilustración 12: Clase Person .....	38
Ilustración 13: Clase Activity .....	39
Ilustración 14: Leyenda del Diagrama RDF.....	39
Ilustración 15: Diagrama RDF .....	40
Ilustración 16: Diagrama de Taxonomía .....	42
Ilustración 17: Jerarquía completa de clases de la ontología .....	50
Ilustración 18: Jerarquía desglosada de clases de la ontología .....	51
Ilustración 19: Creación de Object Properties .....	51
Ilustración 20: Jerarquía de propiedades de objetos.....	52
Ilustración 21: Característica property calls.....	53
Ilustración 22: Característica property consumptionOf.....	53
Ilustración 23: Característica property hasCommand .....	53
Ilustración 24: Característica property hasConsumption .....	54
Ilustración 25: Característica property hasEP .....	54
Ilustración 26: Característica property hasDimmerLampEP .....	55
Ilustración 27: Característica property hasSimpleLampEP .....	55
Ilustración 28: Característica property hasFunctionality .....	56
Ilustración 29: Característica property hasLamps.....	56
Ilustración 30: Característica property hasLight .....	57
Ilustración 31: Característica property hasNaturalLightCondition .....	57
Ilustración 32: Característica property has Notification .....	58
Ilustración 33: Característica property hasOccupant.....	58
Ilustración 34: Característica property hasPrecondition .....	59
Ilustración 35: Característica property hasPreference .....	59
Ilustración 36: Característica property hasState.....	60
Ilustración 37: Característica property highArtificialIlluminationIlluminates.....	60
Ilustración 38: Característica property isCalledBy .....	61
Ilustración 39: Característica property isIlluminatedByHighArtificialIllumination.....	61
Ilustración 40: Característica property isIlluminatedByMediumArtificialIllumination .....	62
Ilustración 41: Característica property isIlluminatedByNoArtificialIllumination .....	62
Ilustración 42: Característica property isIn .....	63
Ilustración 43: Característica property locatedIn.....	63

Ilustración 44: Característica property mediumArtificialIlluminationIlluminates .....	64
Ilustración 45: Característica property necessaryLamps .....	64
Ilustración 46: Características property sensingNaturalLight .....	65
Ilustración 47: Jerarquía de datos de la ontología en Protegé .....	65
Ilustración 48: axioma MusicAtNight .....	66
Ilustración 49: Axioma ReadAtNight .....	66
Ilustración 50: Axioma Sleep .....	66
Ilustración 51: Axioma StudyAtNight .....	66
Ilustración 52: Axioma Device .....	67
Ilustración 53: Axioma Controllable .....	67
Ilustración 54: Axioma Lamp .....	67
Ilustración 55: Axioma DimmerLamp .....	68
Ilustración 56: Axioma OffDimmerLamp .....	68
Ilustración 57: Axioma OnDimmerLamp .....	69
Ilustración 58: Axioma HighIntensityDimmerLamp .....	69
Ilustración 59: Axioma LowIntensityDimmerLamp .....	70
Ilustración 60: Axioma MediumIntensityDimmerLamp .....	70
Ilustración 61: Axioma SimpleLamp .....	71
Ilustración 62: Axioma OffSimpleLamp .....	71
Ilustración 63: OnSimpleLamp .....	72
Ilustración 64: Axioma ShutterActuator .....	72
Ilustración 65: Axioma Sensor .....	73
Ilustración 66: Axioma NaturalLightSensor .....	73
Ilustración 67: Axioma Timer .....	74
Ilustración 68: Axioma HighArtificialIllumination .....	74
Ilustración 69: Axioma LowArtificialIllumination .....	75
Ilustración 70: Axioma MediumArtificialIllumination .....	75
Ilustración 71: Axioma NotArtificialIllumination .....	76
Ilustración 72: Axioma HighNaturalIllumination .....	76
Ilustración 73: Axioma LowNaturalIllumination .....	77
Ilustración 74: Axioma MediumNaturalIllumination .....	77
Ilustración 75: Axioma NotNaturalIllumination .....	78
Ilustración 76: Axioma ControlFunctionality .....	78
Ilustración 77: Axioma NotificationFunctionality .....	79
Ilustración 78: Axioma NocturnMusicListener .....	79
Ilustración 79: Axioma NocturnReader .....	80
Ilustración 80: Axioma NocturnStudent .....	80
Ilustración 81: Axioma HighIntensityContinuousState .....	81
Ilustración 82: Axioma LowIntensityContinuousState .....	81
Ilustración 83: Axioma MediumIntensityContinuousState .....	82
Ilustración 84: Axioma OffContinuousState .....	82
Ilustración 85: Axioma DiscreteState .....	83
Ilustración 86: Axioma OffDiscreteState .....	83
Ilustración 87: Axioma OnDiscreteState .....	84
Ilustración 88: Axioma ShutterState .....	84

Ilustración 89: Regla de Inferencia MusicAtNight.....	85
Ilustración 90: Regla de Inferencia ReadAtNight .....	85
Ilustración 91: Regla de Inferencia Sleep .....	85
Ilustración 92: Regla de Inferencia StudyAtNight .....	85
Ilustración 93: Creación Individuals - Occupant.....	86
Ilustración 94: Creación property age para Pepe.....	87
Ilustración 95: Property de Occupant Pepe .....	87
Ilustración 96: Escoger clase perteneciente a Pepe.....	88
Ilustración 97: Menú Principal Razonador Jena .....	88
Ilustración 98: Ingreso de la Ontología .....	89
Ilustración 99: Ingreso exitoso ontología a razonador.....	89
Ilustración 100: Sentencia general .....	90
Ilustración 101: Resultado Sentencia general.....	90
Ilustración 102: Music .....	91
Ilustración 103: Read.....	91
Ilustración 104: Study.....	91
Ilustración 105: Query llamar todas acciones .....	92
Ilustración 106: Resultado Query todas acciones .....	92
Ilustración 107: Query sujeto padre MusicAtNight .....	93
Ilustración 108: Resultado Query MusicAtNight.....	93
Ilustración 109: Creación Occupant1 .....	94
Ilustración 110: Creación Location1 .....	95
Ilustración 111: Creación LightIntensity1.....	95
Ilustración 112: Creación Music1 .....	96
Ilustración 113: Creación Regla de Inferencia Music1 .....	96
Ilustración 114: Antes ejecución Music1.....	96
Ilustración 115: Después ejecución Music1 .....	97
Ilustración 116: Creación de Inferencia Music1.....	97
Ilustración 117: Creación Occupant2 .....	98
Ilustración 118: Creación Location2 .....	98
Ilustración 119: Creación LightIntensity2.....	99
Ilustración 120: Creación Sleep .....	99
Ilustración 121: Creación Regla de Inferencia Sleep .....	100
Ilustración 122: Antes de la ejecución Sleep.....	100
Ilustración 123: Después ejecución Sleep .....	101
Ilustración 124: Creación de inferencias Sleep .....	101
Ilustración 125: Creación Occupant3 .....	102
Ilustración 126: Creación Location3 .....	102
Ilustración 127: Creación LightIntensity3 .....	103
Ilustración 128: Creación Read .....	103
Ilustración 129: Creación Regla de Inferencia Read.....	104
Ilustración 130: Antes ejecución Read .....	104
Ilustración 131: Después ejecución Read.....	105
Ilustración 132: Creación de inferencias Read .....	105
Ilustración 133: Creación Occupant4 .....	106

Ilustración 134: Creación Location4 .....	106
Ilustración 135: Creación LightIntensity4.....	107
Ilustración 136: Creación Study .....	107
Ilustración 137: Creación Regla de Inferencia Study.....	107
Ilustración 138: Antes ejecución Study .....	108
Ilustración 139: Después ejecución Study.....	108
Ilustración 140: Creación regla de inferencia Study .....	109
Ilustración 141: Reglas de Inferencia .....	110

## ÍNDICE DE TABLAS

Tabla 1: Glosario de Términos.....	41
Tabla 2: Tabla de Relaciones Binaria .....	43
Tabla 3: Descripción de Relaciones Binarias .....	44
Tabla 4: Atributos .....	47
Tabla 5: Condiciones necesarias de algunas clases de la ontología.....	48
Tabla 6: Condiciones suficientes de algunas clases de la ontología .....	49
Tabla 7: Datos de prueba para el Ocupante.....	86
Tabla 8: Comparación de Número de Inferencias Acertadas.....	110
Tabla 9: Fuente de Datos Históricos – Indicador de Inferencias Acertadas.....	111

## CAPITULO I: INTRODUCCION

### 1.1. Planteamiento del problema

Si bien el consumo eléctrico no forma parte del índice de calidad de vida, como lo son la esperanza de vida, el PBI (Producto Bruto Interno) per cápita, la estabilidad política y la tasa de desempleo (ECONOMIST INTELLIGENCE UNIT (EIU), 2004), es un reflejo del nivel de ingreso que posee una persona o familia. Asimismo, ayuda a satisfacer necesidades básicas como la alimentación, vivienda, salud y educación, ya que estas necesidades requieren el uso de energía eléctrica en su procesamiento, construcción, preservación e iluminación.

Pero el lado oscuro de las actividades eléctricas como la generación, transmisión, distribución y el consumo mismo por parte de los clientes, generan impactos ambientales considerables. La quema de combustibles fósiles para la producción eléctrica, la modificación de ecosistemas para construcción de presas y centrales eléctricas, la instalación de torres de tensión y el tendido de redes para el transporte de la electricidad y el impacto del consumo de energía son algunos de ellos.

Entre las actividades eléctricas señaladas la que produce mayores impactos es la generación, en especial cuando la operación de la misma depende de combustibles fósiles.

La energía eléctrica es una energía poco nociva para el medio ambiente pero la construcción de centrales eléctricas y sus operaciones son las que generan el mayor impacto. Aparte de las emisiones propias de la quema de combustible en las centrales eléctricas, existe el ruido que producen las mismas y también, algunos residuos sólidos.

La transmisión y distribución de la energía eléctrica tienen impactos. Estos tienen alcance sobre la fauna de aves, la generación de campos electromagnéticos, además de otros de orden visual, como, por ejemplo, el impacto estético de las redes de tensión sobre patrimonios naturales, históricos y culturales (Chamochumbi, 2013).

Pero este no es el único aspecto preocupante en los peruanos, además de ser considerado actualmente un servicio elevado de precio.

Según artículo de prensa del Diario Perú21 (Perú21, 2016), informan que OSINERGIM aumentarán tarifas eléctricas desde el día 04 de enero del 2016, donde se establece que las tarifas se incrementarán en promedio 2.8% para los usuarios residenciales y 3.9% para los usuarios comerciales e industriales. Si un usuario doméstico pagaba 50 soles mensuales en su recibo de luz, a partir del mes de enero del 2016 este monto se incrementará aproximadamente en 1.40 soles.

De la misma manera el diario El Comercio (El Comercio, 2016), con el encabezado denominado “¿Qué pagamos cuando pagamos el recibo de electricidad?”, informan en porcentajes el aumento que se viene dando en el servicio eléctrico, comentándolo de esta manera, el año pasado los recibos de electricidad aumentaron 18,7% y este año la tendencia continúa ¿se podrá parar?.

Como se puede mostrar con la información presentada, en el Perú los servicios eléctricos son costosos y generan contaminación.

Así, la empresa PROARCO S.A.C. en Trujillo pretende construir edificios multifamiliares con departamentos domóticos, pero el ahorro de energía que estos dispositivos generan es únicamente reflejado en una reducción al 30% según Schneider Electric, haciendo inviable la justificación económica de la inversión de domotizar un departamento ante el poco ahorro energético generado, según entrevista de un consultor de PROARCO S.A.C.

Entonces la empresa PROARCO S.A.C. busca construir departamentos domotizados en su proyecto denominado “Residencial Los Jazmines” adquirido de la empresa BCN Asesores Inmobiliarios S.A.C. que cuenta con un departamento por nivel de 130 m<sup>2</sup> aproximadamente, tal y como se muestra mediante planos en el anexo Nro. 01, pero con la intención de que el controlador de los dispositivos domóticos sea más “Inteligente” mediante tecnología de la Web Semántica como las ontologías que permiten crear

reglas de inferencia. Según entrevista al consultor de PROARCO S.A.C (S.A.C, 2016).

Algunas investigaciones aseguran que con el uso de sistemas o dispositivos inteligentes en los inmuebles se puede reducir el uso de la energía eléctrica. Según la industria Schneider Electric (Schneider Electric, 10), proponen un sistema de administración de energía en los diferentes aparatos eléctricos que existan en diferentes tipo de inmuebles, en el caso de edificios exponen que las instalaciones de climatización e iluminación de los edificios y viviendas consumen gran parte de la energía eléctrica demandada en las ciudades.

Un edificio “inteligente” es una inversión rentable frente a las soluciones tradicionales.

Puede ahorrar a largo plazo hasta un 30% de los gastos habituales. Disponen de soluciones que pueden mejorar la eficiencia de estos sistemas, así como sistemas de gestión de edificios (BMS) que pueden controlar la calefacción y el aire acondicionado para mantener el edificio a una temperatura confortable, minimizando al mismo tiempo la energía utilizada. Cuentas con Sistemas de climatización y ventilación primario, Sistemas contra incendios, Control accesos, CCTV- circuito de video vigilancia, Distribución de potencia, etc.

Finalmente para este caso afirman que hasta un 30% es el ahorro energético que se puede lograr con la implementación de sistemas de gerenciamiento de edificios.

Otros sistemas y dispositivos que contribuyen con el ahorro energético (Hrenhata, s.f.), son los controladores de iluminación, la instalación de dispositivos como programación horaria, sensor de luz de día, detección de presencia, entre otros. Proponen diciendo que si se incorporan en nuestras viviendas los sistemas recomendados para el control de la iluminación, además de obtener confort, se podría conseguir ahorros de entre 0 y 60% en iluminación.

Existen diferentes dispositivos y sistemas que sirven como medios para el ahorro energético, pero si pudimos percatarnos con lo expuesto anteriormente que para lograr este ahorro se necesita la implementación y el uso de diferentes dispositivos, para un fin en común y como resultado los costos serán elevados para poder conseguir el ahorro.

En Trujillo al año 2016 la venta de departamentos se ha reducido (La Industria, 2016), debido al bajo poder de endeudamiento y poca aprobación de crédito hipotecarios por parte de la Banca.

Se desea vender departamentos, pero se necesita dar un valor agregado, una alternativa podría ser la domótica, pero es necesario justificar el incremento en costo de dispositivos e instalaciones de estos en un departamento con el ahorro de energía eléctrica que generarían.

## 1.2. Delimitación del problema:

La presente investigación queda delimitada al desarrollo de una ontología de patrón de comportamiento para anticiparse a las preferencias de usuario en el departamento 301 del proyecto Residencial Los Jazmines de la empresa PROARCO S.A.C., Trujillo año 2016.

## 1.3. Características problemáticas:

- El ahorro de energía de un departamento domotizado no justifica dicha inversión.
- El controlador del sistema Domótico no cuenta con la capacidad de anticiparse a las necesidades del usuario en un departamento domótico.

El ahorro de energía de un departamento domotizado no justifica dicha inversión, pues reduce en un 30% el consumo eléctrico, pero genera un incremento en su costo de S/. 10, 250.00 soles siendo capaz de retornar dicha inversión en más de doce años, haciéndolo inviable. (Urrelo, "Ontología de Patrón de Comportamiento para Incrementar el Ahorro de Energía Eléctrica Buscando Justificar la Inversión en Departamentos Domotizados").

El controlador de un departamento domotizado no cuenta con la capacidad de anticiparse a las necesidades para reducir los momentos de consumo de Kw por dispositivos en stand by.

#### 1.4. Formulación del problema:

En tal motivo existe la necesidad de desarrollo un controlador con mayor capacidad de inferencia, que permita anticiparse a las necesidades de los usuarios y que permite reducir el consumo de energía eléctrica.

Una manera de reducir el consumo de energía eléctrica sería anticipándose en el comportamiento de las personas que habitan un departamento pero ello requeriría un sistema experto empotrado entre los dispositivos domóticas lo cual haría incrementar aún más los costos.

¿Qué componente de Tecnologías de Información permitirá anticiparse a las preferencias de los usuarios en un departamento domotizado del proyecto Residencial Los Jazmines de la empresa PROARCO S.A.C.?

#### 1.5. Formulación de la Hipótesis:

H1: Una Ontología de patrón de comportamiento como controlador Web permitirá anticiparse a las preferencias de los usuarios en el departamento domotizado 301 del proyecto Residencial Los Jazmines de la empresa PROARCO S.A.C. en Trujillo el año 2016.

H2: Una Ontología de patrón de comportamiento como controlador Web no permitirá anticiparse a las preferencias de los usuarios en el departamento domotizado 301 del proyecto Residencial Los Jazmines de la empresa PROARCO S.A.C. en Trujillo el año 2016.

#### 1.6. Objetivos del estudio:

##### 1.6.1. Objetivo General

Desarrollar una ontología de patrón de comportamiento como controlador web de dispositivos en el departamento domotizado 301 del proyecto Residencial Los Jazmines de la constructora PROARCO S.A.C.

### 1.6.2. Objetivos Específicos

- Análisis de los términos y taxonomía de conceptos, de una Ontología con dispositivos de un departamento domotizado y patrón de comportamiento de personas habitantes en dicho departamento según la metodología Methontology y reutilizando conceptos de las ontologías DogOnt y DogPower.
- Definir las relaciones binarias de una Ontología con dispositivos de un departamento domotizado y patrón de comportamiento de personas habitantes en dicho departamento según la metodología Methontology.
- Definir los atributos, axiomas y reglas de inferencia de una Ontología con dispositivos de un departamento domotizado y patrón de comportamiento de personas habitantes en dicho departamento según la metodología Methontology.
- Implementar la Ontología anteriormente desarrollada utilizando el software Protégé y ejecutarla mediante el razonador Pellet.

### 1.7. Justificación del Estudio:

En el aspecto académico, se pretende lograr con esta investigación cumplir con la primera introducción de los conceptos relacionados con el desarrollo de Ontologías de la Web Semántica, así como técnicas y métodos para desarrollarla mediante uso de software como Protégé y Pellet.

En el aspecto social, su importancia radica en brindar una herramienta para complementar la construcción de edificios domóticos, con el fin del ahorro de la energía eléctrica. Así se puede contribuir a reducir la contaminación y consigo el calentamiento global.

Proporcionar una ontología más inteligente, capaz de anticiparse a las preferencias de los usuarios de un ambiente.

Permite brindar una herramienta de reducción de consumo energético y fomentar la construcción de edificaciones responsables con el medio ambiente.

Demostrar a la sociedad una manera en la que el sector construcción y tecnología pueden contribuir en el cuidado del medio ambiente, sin dejar de priorizar sus actividades fundamentales, como satisfacer las necesidades de vivienda y facilitar la vida con el soporte tecnológico.

Generar en los estudiantes de tecnología la iniciativa de explorar nuevos conceptos, metodologías, técnicas; para convertir TI en apoyo a la calidad de vida de las personas.

Una posibilidad de cuidar el medio ambiente en la reducción del servicio eléctrico, con apoyo de la tecnología y la construcción.

En demostrar que se puede mejorar la satisfacción del usuario de un departamento domotizado de la empresa PROARCO S.A.C.

De manera holística la sociedad que accedan a la adquisición de estas viviendas.

Conocedores, constructores y tecnológicos.

Los usuarios de los departamentos domóticos, tendrán un ahorro eléctrico consigo un ahorro financiero del departamento a través de un dispositivo.

A los conocedores se benefician con la generación de esta innovación y conocimiento tecnológico aplicado a la construcción.

#### 1.8. Limitaciones del Estudio:

Limitación del tiempo del PADT de cuatro meses con la necesidad de profundizar en el conocimiento para el desarrollo de Ontologías de la Web Semántica y tecnologías como: Protégé y Pellet.

## CAPITULO II: MARCO TEÓRICO

### Antecedentes

Santiago Moreno Carbonell (2014), en su tesis “Interfaz para el Control de un Entorno Doméstico Mediante la Voz” demostró la factibilidad para conectar diversos dispositivos en un entorno domotizado a través de una ontología y con una interfaz capaz de dar respuesta a consultas complejas y llevar a cabo comando sobre la ontología correctamente, respondiendo al usuario mediante

lenguaje natural hablado, mediante el reconocedor Google Speech Recognition y el sintetizador de voz Google Text To Speech. Se comprobó el procesamiento de los comandos y consultas obteniendo una baja y nula ambigüedad para los comandos y consultas, respectivamente. Sin embargo, en cuanto a los tiempos de respuesta de los razonadores semánticos tanto para dispositivos Android y ordenadores personales, existe una gran desventaja para los primeros, llegando a demorar desde medio segundo para las consultas simples, hasta un máximo de 26s.

(Carbonell, 2014)

Leticia Ramos (2014), en su tesis de postgrado “Uma Proposta de Ontologia para Residências Inteligentes Buscando a Integração de Dispositivos”, desarrolló un modelo ontológico para una residencia denominada “casa inteligente”, con la finalidad de integrar los dispositivos heterogéneos, además de tener la capacidad de adquirir y aplicar conocimiento sobre el ambiente y automáticamente adaptarlo a sus habitantes, lo cual resulta ser un enfoque de amplio espectro aplicable en gran parte a la presente tesis.

(Ramos, 2014)

Luis Vladimir Urrelo Huiman (2016), en su artículo “Ontología de Patrón de Comportamiento para Incrementar el Ahorro de Energía Eléctrica Buscando Justificar la Inversión en Departamentos Domotizados” desarrolló la simulación del desempeño de una ontología sobre un departamento domotizado para determinar la existencia de un ahorro en lo que respecta al consumo eléctrico, para ello, usó los conceptos de las ontologías DogOnt y DogPower, pasando a probar el resultado de esta combinación con la librería Jena, siguiendo la metodología Methontology en el software Protégé, concluyendo con la medición del tiempo de activación de actuadores simulados identificando el factor de ecuación diferencial mediante Matlab.

(Urrelo, 2016)

Dario B., Fulvio C., Faisal R. (2010), en su artículo "Enabling machine understandable exchange of energy consumption information in intelligent domotic environments" desarrollaron un Marco de trabajo para la publicación de

información energética, semánticamente (Semantic Energy Information Publishing Framework). Determinando escenarios, tales como: Sistema de administración de energía, el cual consistía en un dispositivo de tipo plug andplay capaz de informar sobre el consumo de energía de los dispositivos y sugerir mejores planes de energía; Redes de energía inteligente 2020, el cual es una proyección al consumo energético de la tierra y el uso de la web semántica y conceptos de la ontología DogOnt para su aplicación a la domótica.

(Bonino, Corno, & Razzak, 2011)

## 2.1. Bases teóricas

### 2.1.1. Departamentos domotizado y el patrón de comportamiento de los usuarios.

La domótica en departamentos aplica una serie de técnicas orientadas a automatizar las acciones cotidianas que una persona lleva a cabo en su hogar, tales como: encender las luces, programar artefactos, permitir acceso a diversos ambientes, etc.

El patrón de comportamiento de los usuarios permite volver a una casa domotizada en una casa inteligente, gracias a que el sistema llega a tal nivel de madurez que pasa casi desapercibida debido a que la misma interacción entre el usuario y sistema genera una base de funcionalidades que se pueden adaptar a las circunstancias del día.

### 2.1.2. Desarrollo de Ontologías de la Web Semántica.

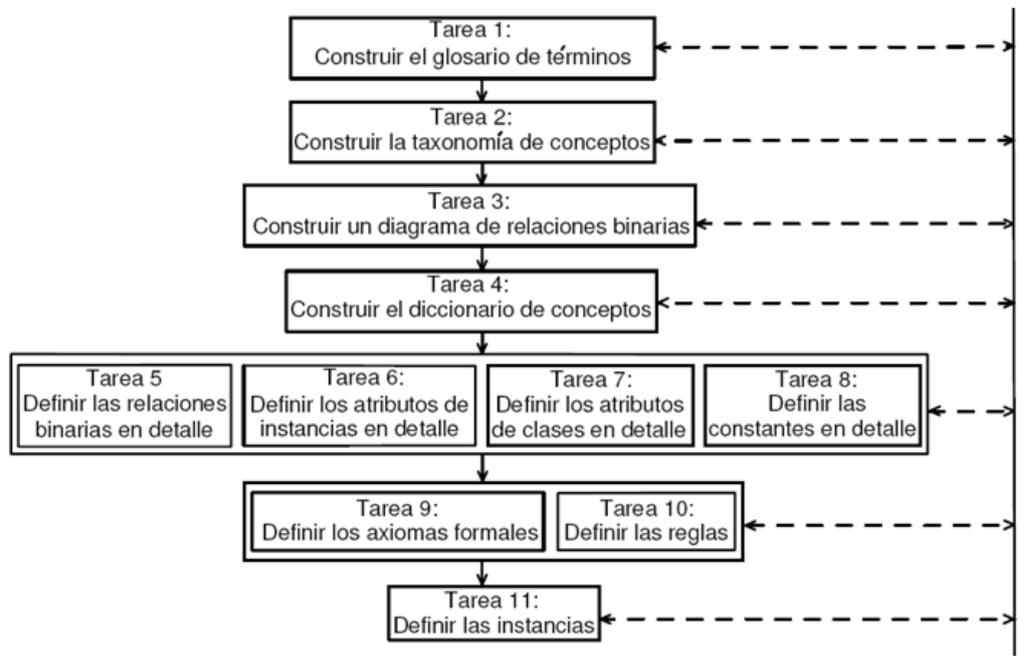
La ontología de web semántica es la manera en que la web semántica debe de estar codificada para su interpretación por los ordenadores sin intervención del hombre.

### 2.1.3. Methontology.

Es una metodología para elaborar ontologías en el nivel de conocimientos, se basa en las actividades identificadas por el proceso de desarrollo de software propuesto por la IEEE (Instituto de Ingeniería Eléctrica y Electrónica) y en otras metodologías de ingeniería de conocimientos, estas son las siguientes:

- La especificación permite determinar por qué se construye la ontología, cuál será su uso, y quienes serán sus usuarios finales.
- La conceptualización se encarga de organizar y convertir una percepción informal del dominio en una especificación semi-formal, para lo cual utiliza un conjunto de representaciones intermedias, basadas en notaciones tabulares y gráficas, que pueden ser fácilmente comprendidas por los expertos de dominio y los desarrolladores de ontologías.
- La formalización se encarga de la transformación de dicho modelo conceptual en un modelo formal o semi-computable.
- La implementación construye modelos computables en un lenguaje de ontologías (Ontolingua, RDF Schema, OWL, etc.) La mayor parte de las herramientas de ontologías permiten llevar a cabo esta actividad de manera automática.
- El mantenimiento se encarga de la actualización y/o corrección de la ontología, en caso necesario.

Ilustración 1: Modelado Conceptual de una Ontología



(Corcho O)

#### 2.1.4. Protégé y Jena

Protégé es un editor libre de código abierto y un sistema de adquisición de conocimiento. Al igual que Eclipse, Protégé es un framework para el cual otros proyectos sugieren plugins donde la aplicación estaría escrita en Java y trabajaría fuertemente con Swing para crear su compleja interfaz.

Apache Jena es un framework de código abierto de Web Semántica para Java. Proporciona una API para extraer datos desde y escribir en gráficos RDF. Los gráficos se representan como un "modelo" a la moda. Un modelo puede ser de origen con datos de archivos, bases de datos, URL o una combinación de estos. Un modelo también se puede consultar a través de SPARQL 1.1.

Jena proporciona soporte para OWL (Web Ontology Lenguaje). El marco tiene varios razonadores internos y el razonador Pellet (un código abierto de Java OWL-DL razonador) se puede configurar para trabajar en Jena.

#### 2.1.5. Ontología:

Es la formulación de un preciso esquema conceptual sobre uno o varios dominios con la misión de agilizar la comunicación y el intercambio de información entre diversos sistemas y entes. (ABC, Definición de Ontología)

En Ciencia y Tecnologías de la Información, las ontologías son clasificaciones. Se utilizan como un medio para categorizar o agrupar la información en clases.

Las ontologías se aplican también en Web Semántica y en Inteligencia Artificial para asimilar y codificar el conocimiento, definiendo las relaciones existentes entre los conceptos de un determinado dominio (un área del conocimiento). (Significados, s.f.)

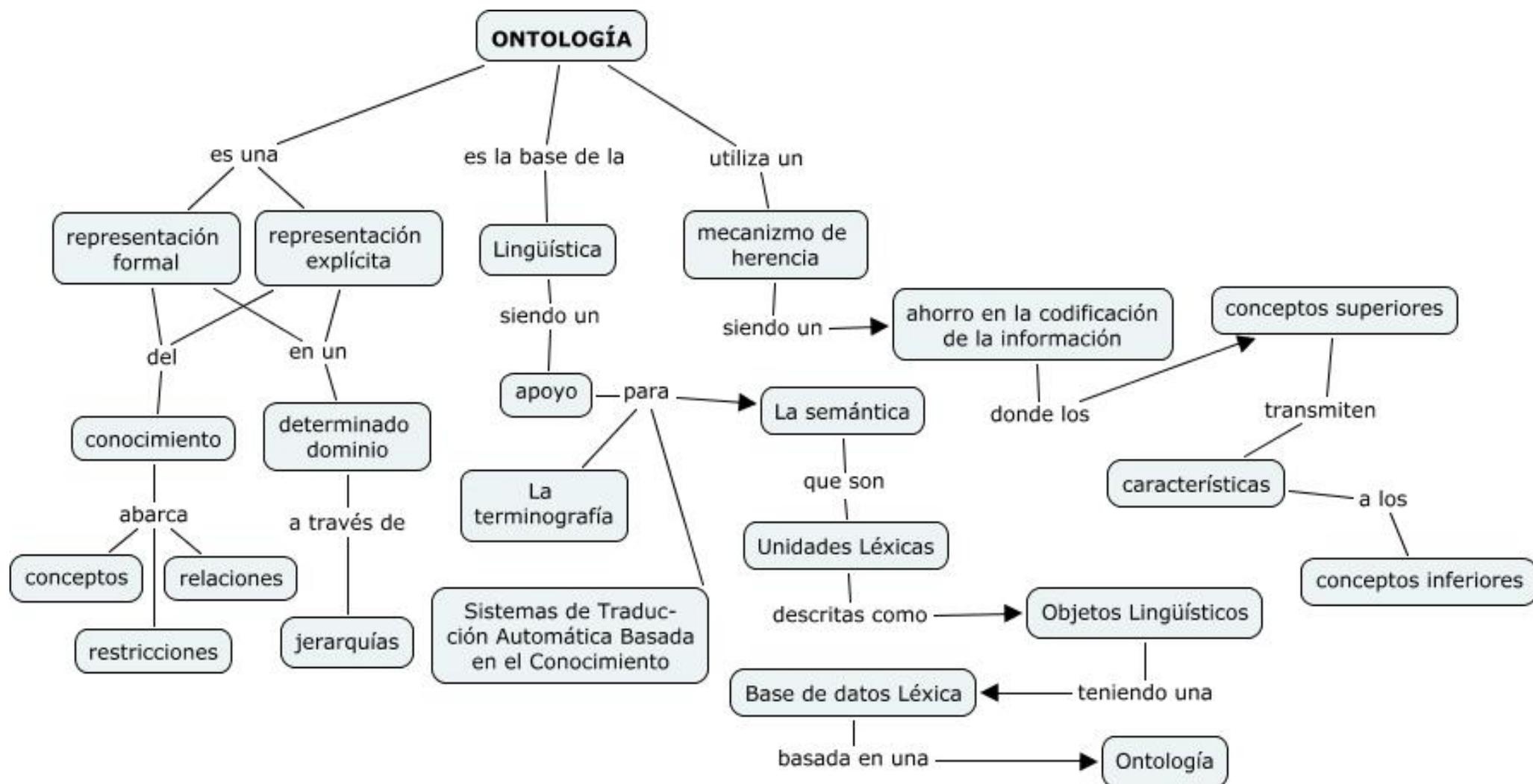


Ilustración 2: La Ontología

(cmapspublic3, s.f.)

#### 2.1.6. Patrón de Comportamiento:

Cuando ciertas reacciones de la persona, se hacen muy frecuentes en determinados ambientes o situaciones, constituyen lo que llamamos un patrón de comportamiento. Un patrón de comportamiento es una forma constante que tiene una persona, de pensar, sentir, reaccionar físicamente y actuar en determinada situación. (González)

#### 2.2.7 Domótica:

Se denomina como domótica a la serie de sistemas integrados e interrelacionados que se instalan en un hogar y que permiten la automatización del mismo y su control tanto desde adentro de la casa como desde afuera. (ABC, Definición de Domótica).

#### 2.2.8 DogOnt:

Es un modelo de ontología diseñado para soportar la interoperación, integración e inteligencia en ambientes domóticos. Posee conceptos tales como:

- Building Thing: Modela todos los elementos del ambiente del edificio (Building Environment) dividido en:
  - Controllable
  - UnControllable
    - Furniture elements.
    - Walls, floors, ceilings and other architectural elements (Architectural sub-tree).
- Building Environment: Modela los cuartos (rooms) y los espacios arquitectónicos que componen la casa.
  - Rooms.
  - External spaces such as garages, garden, etc.
- State: Son clasificados de acuerdo al tipo de valores que pueden asumir:
  - Continuous states: Modela continuamente las cualidades cambiantes. (El nivel actual de la disminución de la luz de una lámpara)

- Discrete states: Modela discretamente las cualidades cambiantes. (La lámpara siendo encendida o apagada).
- **Functionality**
  - Continuous Functionalities: Modela la habilidad para cambiar las propiedades de los dispositivos en una manera continua. (Disminuir la intensidad de la luz de una lámpara).
  - Discrete Functionalities: Modela la habilidad de abruptamente cambiar las propiedades de los dispositivos. (Encender o apagar una lámpara).
- **Domotic Network Component**

(Bonino & Corno, 2008)

Ontología DogOnt que contiene los conceptos de un departamento con un sistema domótico.

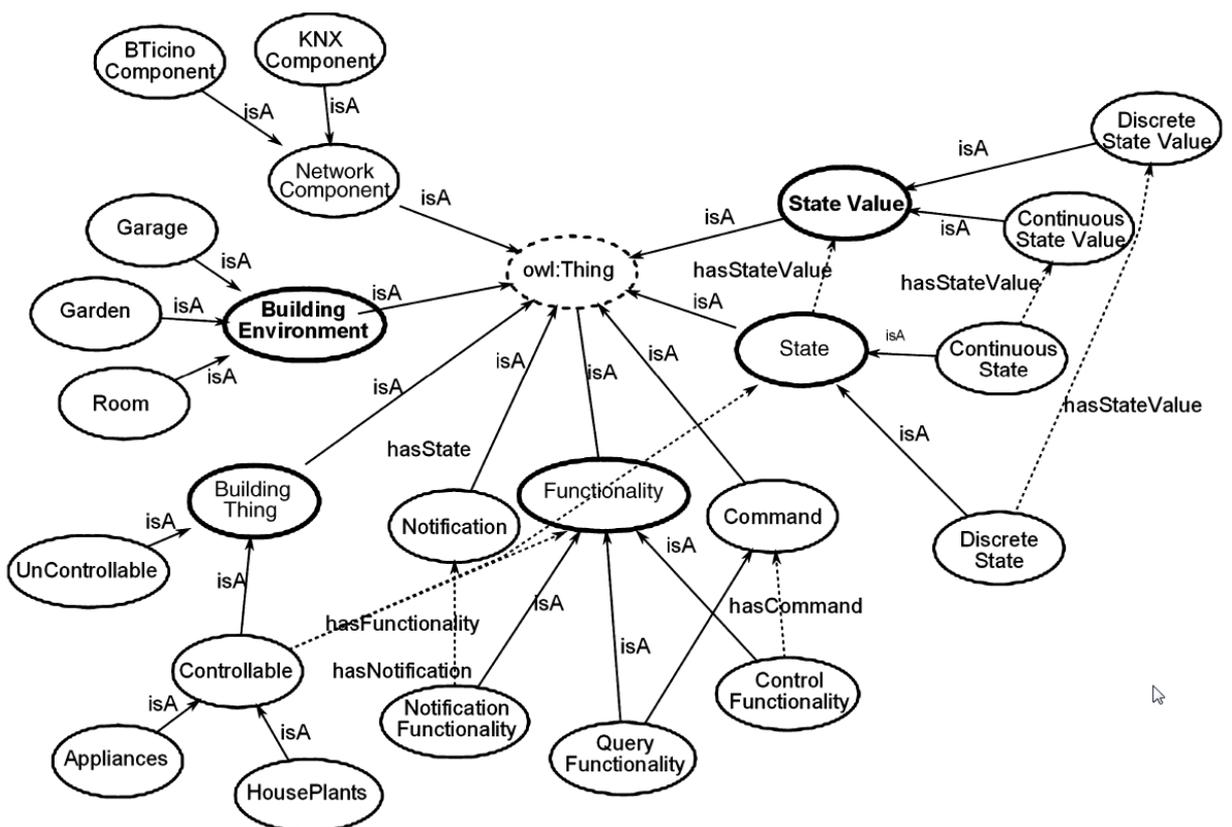


Ilustración 3: DogOnt 2.5.

(Bonino, Corno, & Razzak, 2011)

Ontología DogPower que permite medir el consumo de energía de cada dispositivo de un sistema domótico.

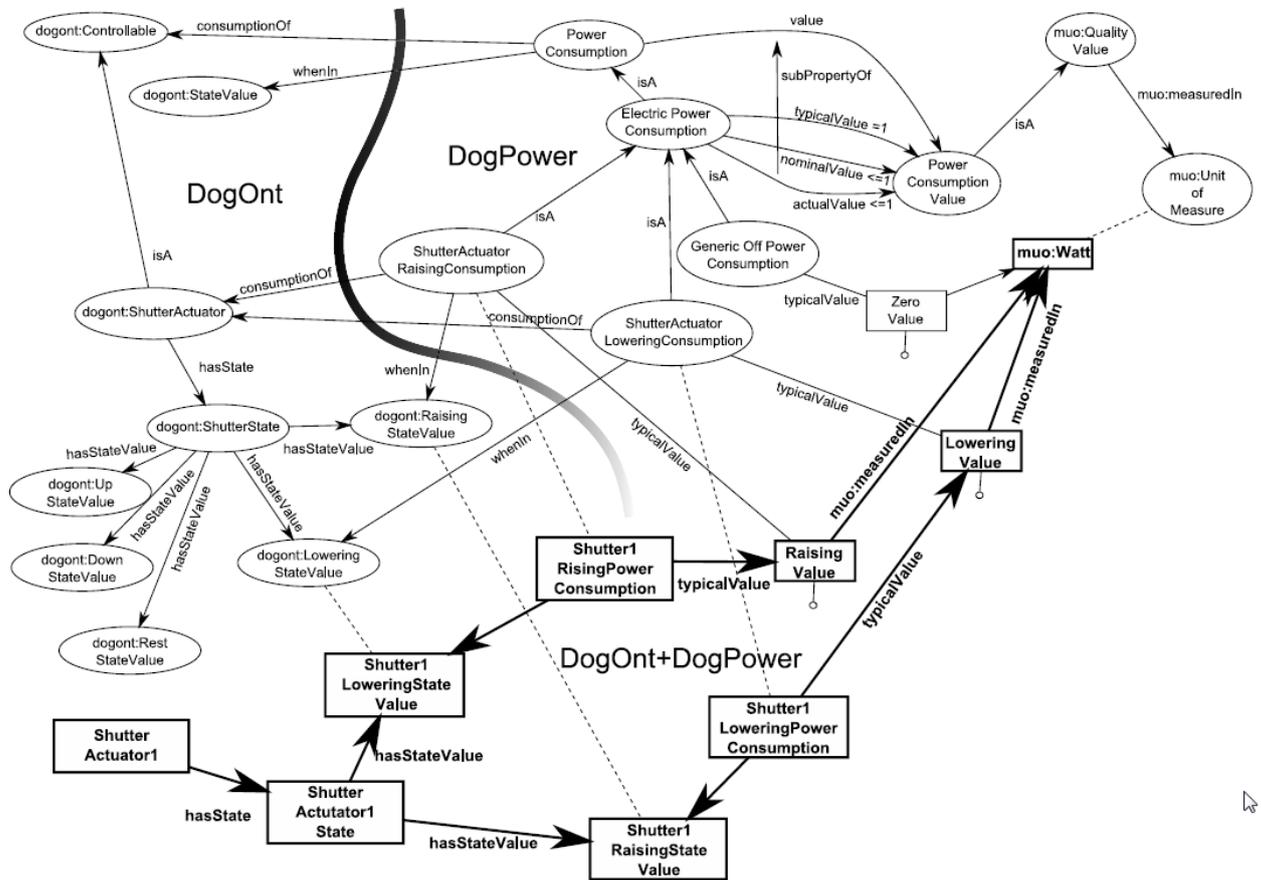


Ilustración 4: Ontologías DogOnt y DogPower

(Bonino D. &., 2014)

### 2.2.9 SPARQL:

Es un lenguaje RDF consulta, es decir, una semántica lenguaje de consulta de bases de datos, capaz de recuperar y manipular los datos almacenados en la Descripción de Recursos (RDF) formato. Se hizo un estándar por el Grupo de Trabajo de Acceso de datos RDF (DAWG) del Consorcio de la World Wide web, y es reconocido como una de las tecnologías clave de la web semántica. El 15 de enero de 2008, SPARQL 1.0 se convirtió en una recomendación oficial del W3C, y SPARQL 1,1 en marzo de 2013.

SPARQL permite una consulta que consiste en patrones triples, conjunciones, disyunciones, y opcionales patrones.

Implementaciones para varios lenguajes de programación existen. Existen herramientas que permiten conectar y construir una consulta SPARQL para un SPARQL endpoint semi-automática, por ejemplo, ViziQuer. Además, existen herramientas que se traducen consultas SPARQL a otra consultar idiomas, por ejemplo, para SQL y para XQuery.

Ventajas:

SPARQL permite a los usuarios escribir consultas en lo que vagamente se puede llamar los datos de valores clave o, más específicamente, los datos que sigue la especificación RDF de la W3C. La base de datos completa es, pues, un conjunto de triples "sujeto-predicado-objeto". Esto es análogo a algunas bases de d

(<https://en.wikipedia.org/wiki/SPARQL>, s.f.)

## CAPITULO III: MATERIAL Y METODOS

### 3.1. Material

#### 3.1.1. Población

Habitantes de la Residencial Los Jazmines.

#### 3.1.2. Muestra

Habitantes del departamento 301 de la Residencial Los Jazmines

#### 3.1.3. Unidad de Análisis

Ontología de Patrón de Comportamiento de habitantes de un departamento domotizado.

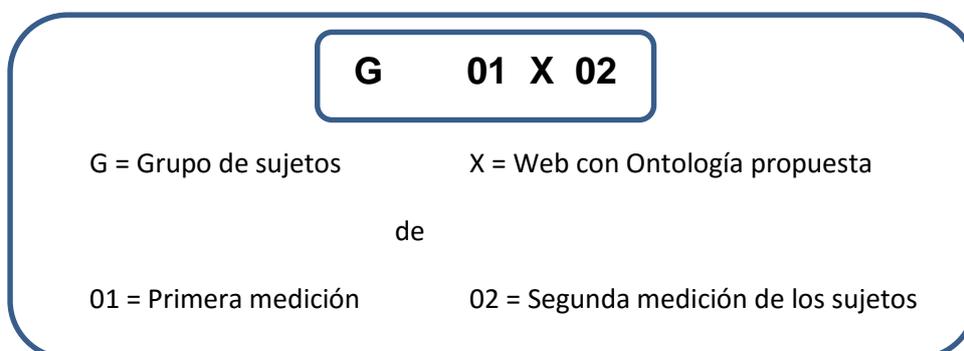
### 3.2. Metodología

#### 3.2.1. Nivel de Investigación

De acuerdo a la naturaleza del estudio de la investigación, reúne por su nivel de características de un estudio descriptivo

#### 3.2.2. Diseño de Investigación

Para el diseño de la investigación, emplearemos el de una investigación por su contratación no experimental. Su diagrama es el siguiente:



#### 3.2.3. Variable de estudio y operacionalización

- Variable Independiente  
Ontología de patrón de comportamiento como controlador Web de dispositivos.
- Variable Dependiente  
Anticiparse a las preferencias de los usuarios en el departamento domotizado 301 del proyecto Residencial Los Jazmines de la empresa PROARCO S.A.C. en Trujillo el año 2016.

### Matriz de Operacionalización de Variables

Variable	Definición conceptual	Indicadores	Unidad de Medida	Instrumento
<b>Dependiente:</b> Anticiparse a las preferencias de los usuarios en el departamento domotizado 301 del proyecto Residencial Los Jazmines de la empresa PROARCO S.A.C. en Trujillo el año 2016.	Son aquellas respuestas obtenidas mediante las reglas de inferencia.	Número inferencias de la iluminación en un determinado ambiente del departamento.	Unidades	Observación
<b>Independiente:</b> Ontología de patrón de comportamiento como controlador Web de dispositivos.	Ontología con conceptos, atributos, relaciones y reglas que permiten generar inferencias, en este caso en relación al patrón de comportamiento de personas ante dispositivos de un departamento domótico, con el fin de anticiparse a sus necesidades y reducir el consumo de energía.	Confiabilidad	Grado de satisfacción.	Prueba de Caja negra.

#### 3.2.4. Técnicas e instrumentos de recolección de datos

Las principales técnicas que se utilizarán en la investigación son:

- Simulación
- Observación
- Reportes
- Encuestas

### 3.2.5. Técnicas de procesamiento y análisis de datos

Para el procesamiento, análisis e interpretación de datos se emplearán los siguientes cálculos estadísticos:

Promedio de las diferencias

$$\bar{d} = \frac{\sum d_i}{n}$$

Desviación estándar de las diferencias

$$S_d = \sqrt{\frac{\sum d_i^2 - \frac{(\sum d_i)^2}{n}}{n-1}}$$

Prueba Wilcoxon para datos emparejados:

$$t_v = \frac{\bar{d}}{\frac{S_d}{\sqrt{n}}} \quad \nu = n - 1 \quad \text{Grados de Libertad}$$

## CAPITULO IV: RESULTADOS

En el presente capítulo se presenta el desarrollo de la Ontología con dispositivos de un departamento domotizado y patrón de comportamiento de personas habitantes en dicho departamento según la metodología Methontology y reutilizando conceptos de las ontologías DogOnt y DogPower.

Para el desarrollo de la ontología se toma en cuenta el uso del software libre Protégé en su versión 5.0, debido a que es la más actualizada y completa.

Las actividades de desarrollo propuestas por la metodología son las siguientes:

a) Especificación.

El propósito de la ontología a desarrollar es simplificar la determinación del patrón de comportamiento de los habitantes en un departamento y lograr una respuesta de habilitación o deshabilitación de actuadores. En cuanto al uso está enfocado al dominio de comportamiento de personas en un departamento domotizado con la aplicación de iluminación, buscando ahorrar el consumo de energía eléctrica. En nuestro caso el escenario de aplicación es un departamento unipersonal de 130 m<sup>2</sup> de la Residencial Los Jazmines del Grupo Proarco SAC., que comúnmente está habilitado por la persona entre 25 y 44 años de edad.

Esta ontología es de carácter semi-formal, debido a la reutilización de conceptos y propiedades de dominios como: DogOnt, DogPower, ADL y la formalización de la ecuación diferencial del consumo de energía de los focos.

### 4.1. Términos y taxonomía de una Ontología con dispositivos de un departamento domotizado y patrón de comportamiento de personas habitantes en dicho departamento.

Es considerado como regla el uso del idioma extranjero inglés, para el desarrollo de ontologías. Con esta consideración se procede con el progreso de las actividades.

La definición del glosario de términos y taxonomía de conceptos de una ontología siguiendo la metodología methontology se encuentran en las etapas iniciales de esta como se observa a continuación:

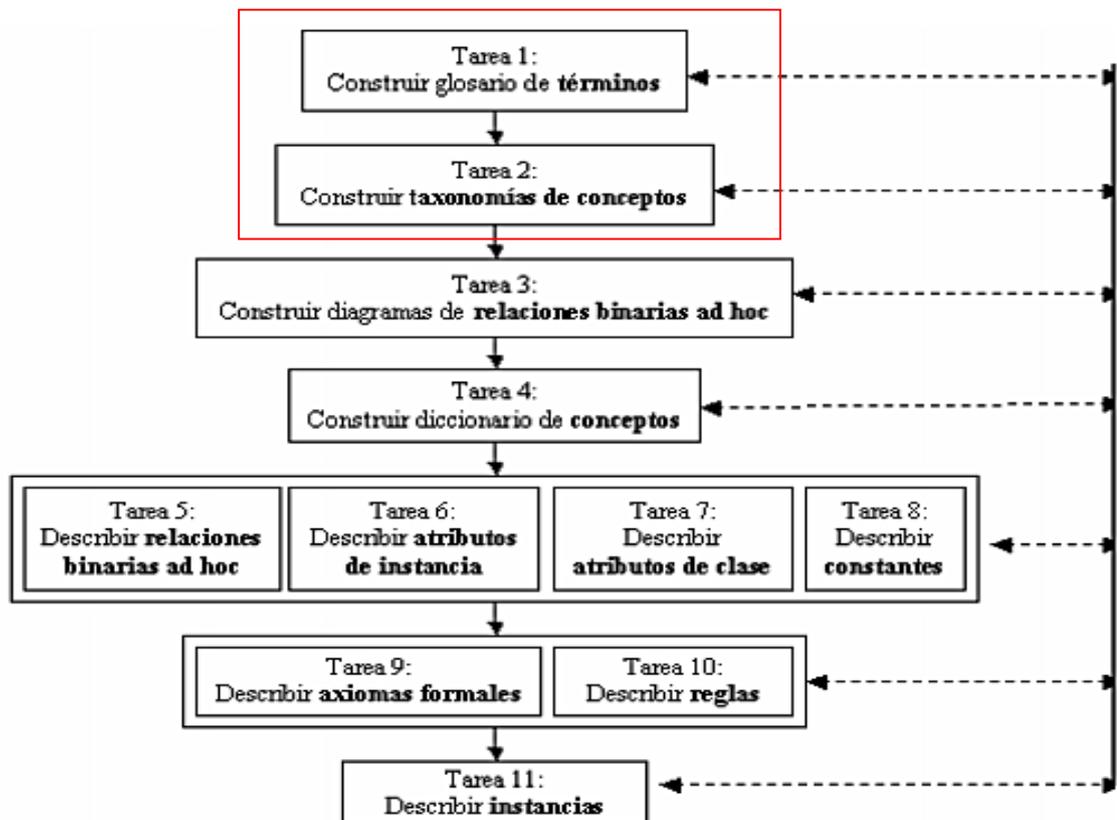


Ilustración 5: Tareas de conceptualización en Methontology

- Tarea 1: Construir el glosario de términos.

Para ello nos basaremos en el grafo RDF del caso describiendo los conceptos relevantes.

Con información e ilustración (Ramos, 2014), se extraen diagramas RDF útiles para el análisis y construcción de la Ontología a desarrollar, para esto hemos seleccionado los siguientes diagramas con sus respectivas adaptaciones para el desarrollo del proyecto enfocado en lumínica.

### Clase BuildingEnvironment

La clase BuildingEnvironment representa los espacios delimitados de una residencia. Gran parte de las ontologías analizadas tiene clases para representar el entorno, ejemplo, DomoML-env (Sommaruga, Perri, & Furfari, 2005) y DogOnt (Bonino & Corno, 2008) tienen la clase BuildingEnvironment, Cobra-Ont (Chen, 2004) posee una clase Lugar. Las clases poseen subclases para clasificar el medio ambiente. Cobra-Ont (Chen, 2004) tiene las subclases AtomicPlaceInBuilding y AtomicPlaceNotInBuilding para identificar si un sitio está ubicado internamente o externamente a una residencia, lo mismo ocurre en la ontología CONON (Wang, 2004) y a través de las subclases OutdoorSpace e IndoorSpace.

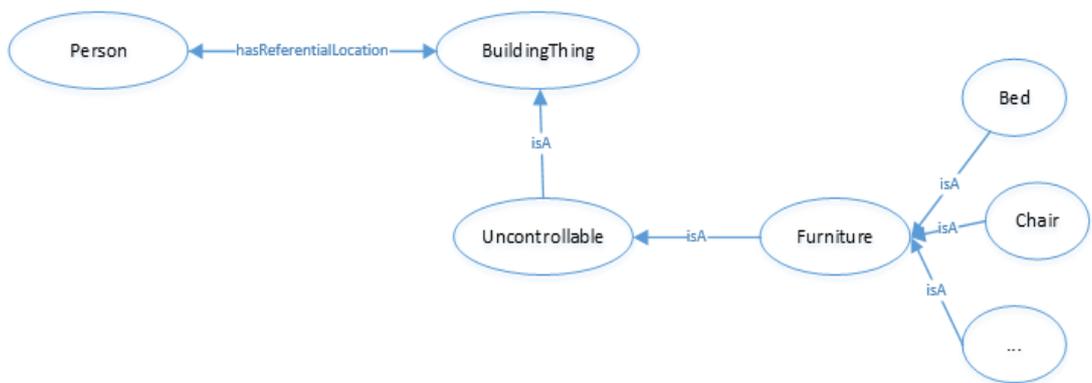


Ilustración 6: Clase BuildingEnvironment

### Clase GeoLocation

La clase de geolocalización, Ilustración 5, le permite establecer una posición de referencia de un objeto o persona en relación con el medio ambiente. Basado en WGS84 (W3C., 2006) es ASB coordenadas longitud, latitud y altitud sobre el nivel del mar a través de las propiedades de los datos con el mismo nombre. La propiedad de objeto hasLocation relaciona una instancia de la clase con individuos de geolocalización de la persona o clase BuildingThing, de esta manera se puede representar la ubicación espacial de una persona o dispositivo.

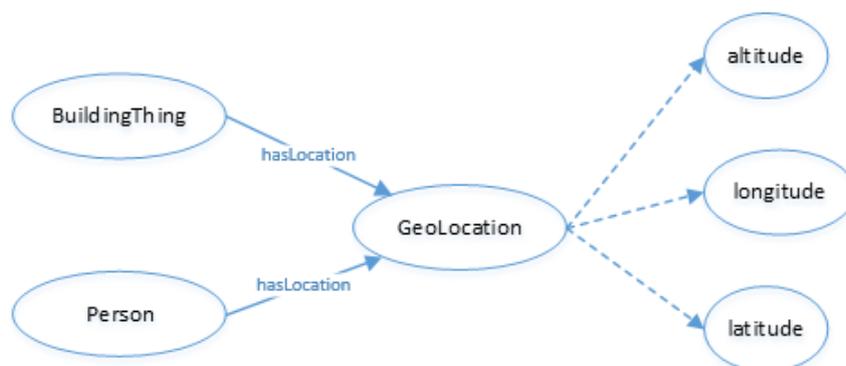


Ilustración 7: Clase GeoLocation

## Clase Controllable

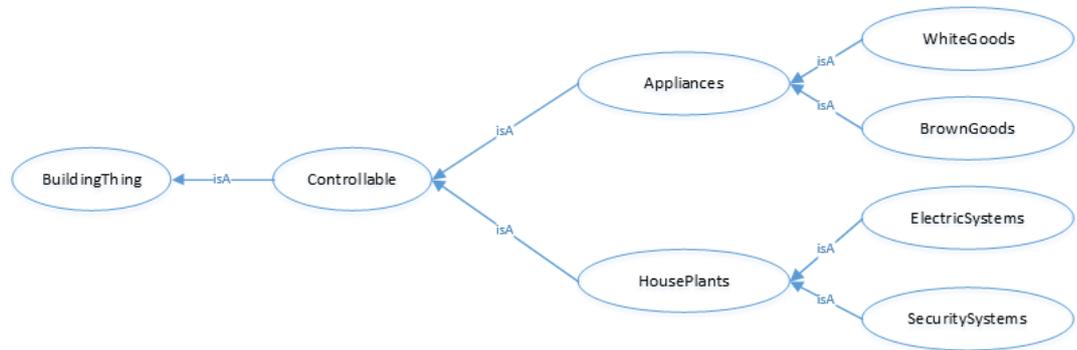


Ilustración 8: Clase Controllable

## Clase Functionality

Una clase de funcionalidad se describe las funciones de automatización de acuerdo con la clase de dispositivos controlables y cada función se define los posibles comandos (Clase Command) para modificar un dispositivo. Las funciones se dividen en diferentes categorías según sus objetivos (Ilustración 7). Los modelos de clase ControlFunctionalities la capacidad de controlar un dispositivo o parte del mismo. La clase NotificationFunctionalities representa la capacidad de una Dispositivo para informar de su estado interno o señal del cambio de un estado. La clase QueryFunctionalities comprende la capacidad de un dispositivo para ser consultados en una condición de fallos o estados.

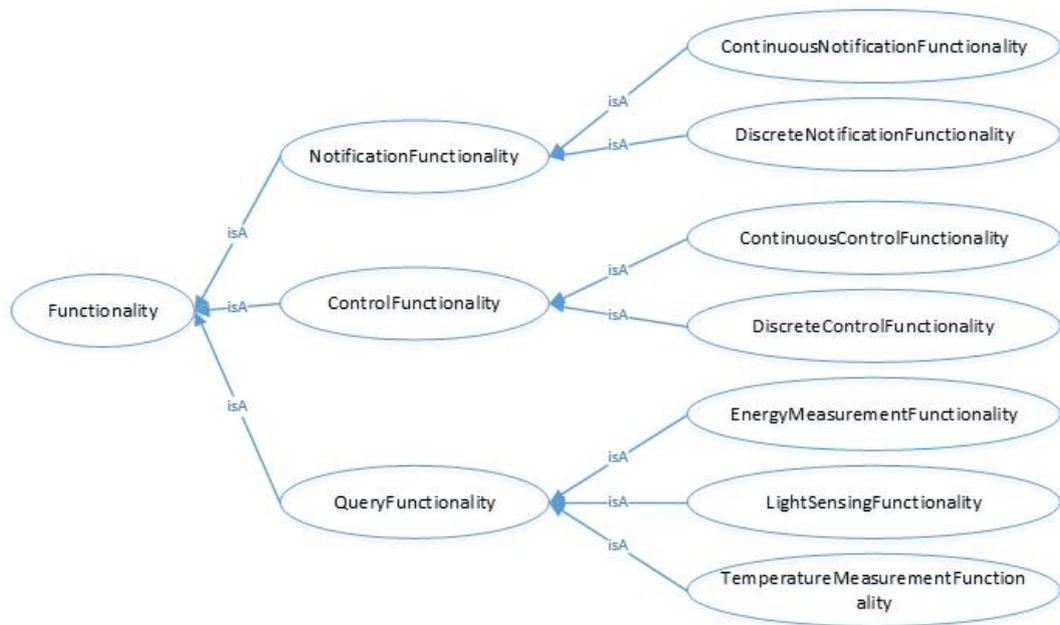


Ilustración 9: Clase Functionality

## Clase Command

La clase de comandos, la ilustración 8 muestra los posibles comandos que una funcionalidad de control o consulta se puede ejecutar. Cada subclase tiene una propiedad de tipo de datos diferente en función del valor de ser representado que puede ser de tipo string o double, por ejemplo. La clase se divide en dos subclases: VoidCommand y NonVoidCommand. La clase VoidCommand corresponde a las funciones que tiene una sola entrada y la clase NonVoidCommand se utiliza para los comandos que devuelven una respuesta

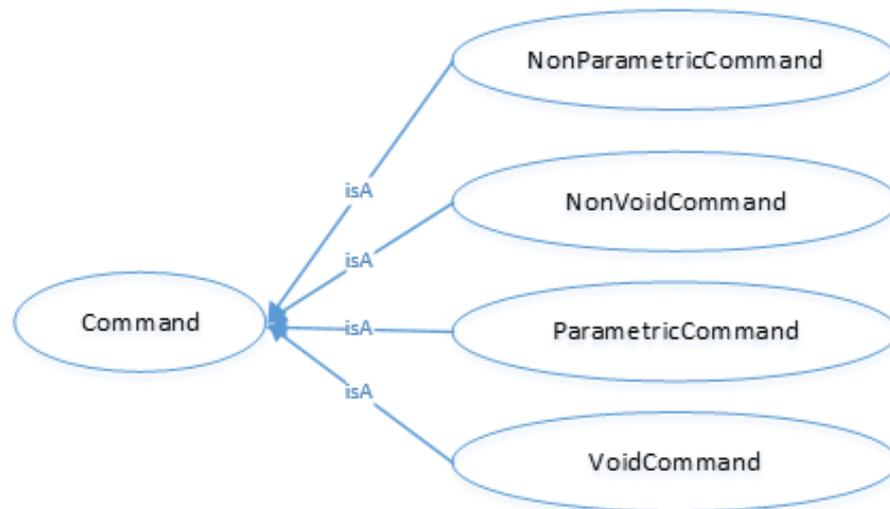


Ilustración 10: Clase Command

## Clase State y StateValue

La clase State, Ilustración 9, clasifica los estados posibles de una clase de dispositivo Controlable. Los Estados se clasifican como continua o discreta de acuerdo con el tipo de valor que puede asumir. Los valores están representados en la clase StateValue y los estados de la clase State tener uno o más estados relativos a la clase StateValue a través de la propiedad hasStateValue. Las subclases de la clase ContinuousValue tener una propiedad de datos de tipo float, mientras que las subclases DiscreteValue tienen un conjunto de dos o tres propiedades de los datos de acuerdo con los valores posibles. Por ejemplo, una clase controlable lámpara tiene el estado que es un estado OnOffState con dos posibles valores representados en la clase StateValue (clases OnStateValue y OffStateValue). Los valores reales de los estados están representados por una propiedad de datos de cadena de tipo "On" u "Off".

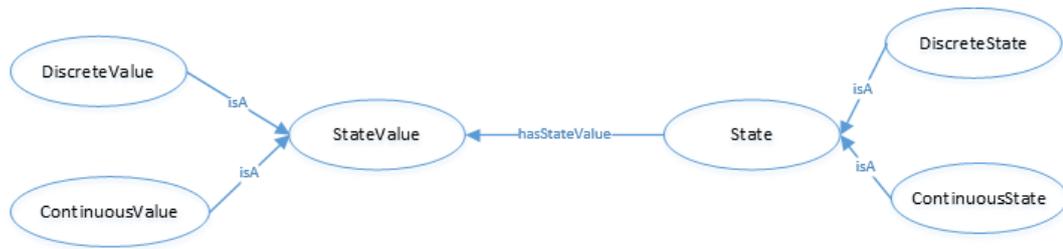


Ilustración 11: Clase State y StateValue

### Clase Person

Una residencia también debe tener en cuenta los habitantes, cada persona tiene sus propias características que pueden influir en el resultado de la interacción. Además de las características individuales, las actividades llevadas a cabo en un momento dado también influyen en el contexto de la automatización. Por último, un usuario tiene preferencias o necesidades que una aplicación debe cumplir. Por lo tanto, cuatro clases fueron creados en relación con la persona: persona, Actividad, Necesidad y FunctionalImpact.

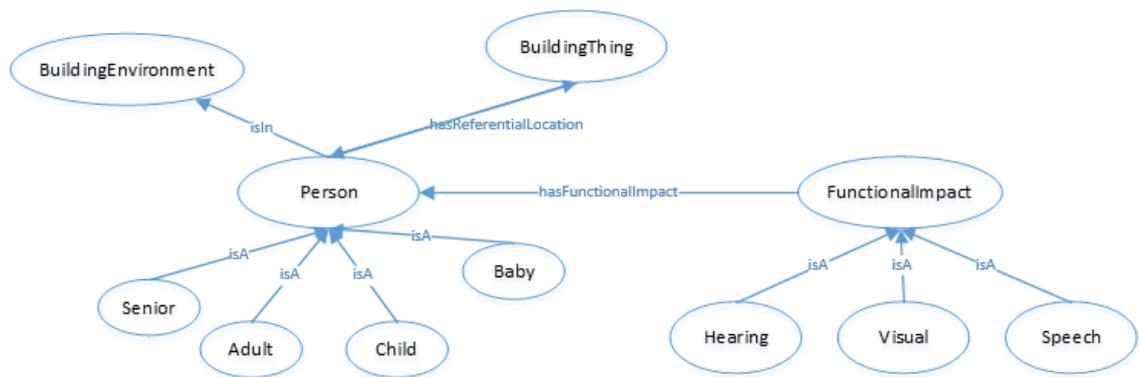


Ilustración 12: Clase Person

## Clase Activity

La Clase Activity, representado en la ilustración 11, se describe las actividades del usuario en dado tiempo. El concepto de actividad es importante para determinar el contexto de aplicación, por ejemplo, si el habitante de la casa está dormido las luces pueden estar apagadas. Se enumeran diversas actividades: leer, caminar, mecanografía, Sentado, comer, hablar, Llamada. La propiedad de objeto performedBy indica la actividad de un usuario.

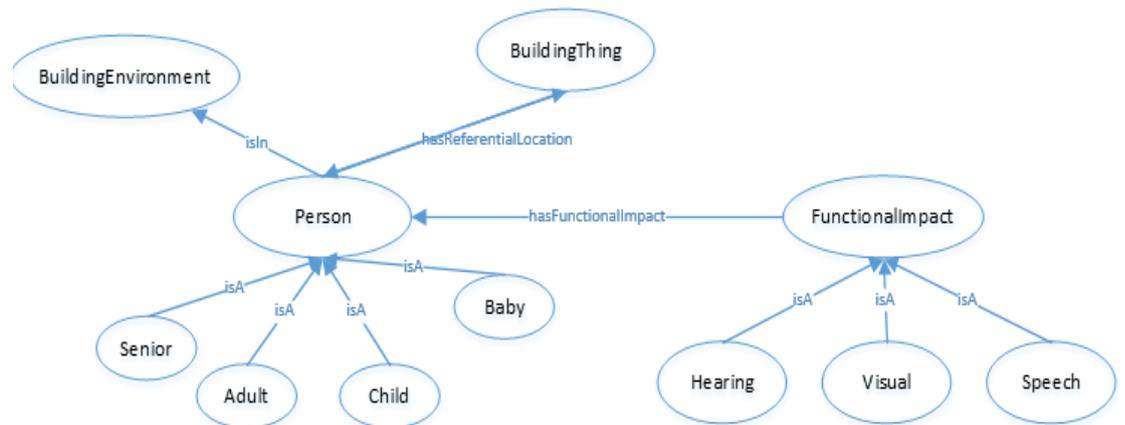


Ilustración 13: Clase Activity

Después de la revisión bibliográfica, el proceso de análisis con las ilustraciones anteriormente mostradas y considerando las limitaciones del desarrollo de la tesis, se ha concluido en el siguiente diagrama RDF para luego realizar el Glosario de Términos.

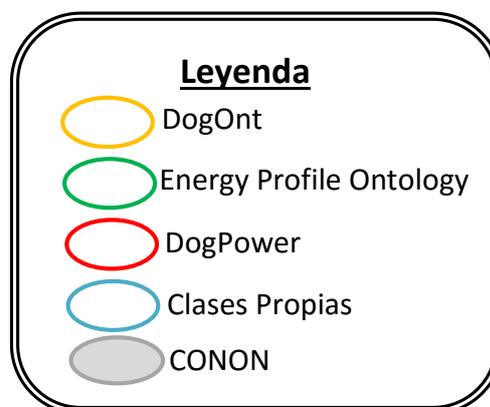


Ilustración 14: Leyenda del Diagrama RDF

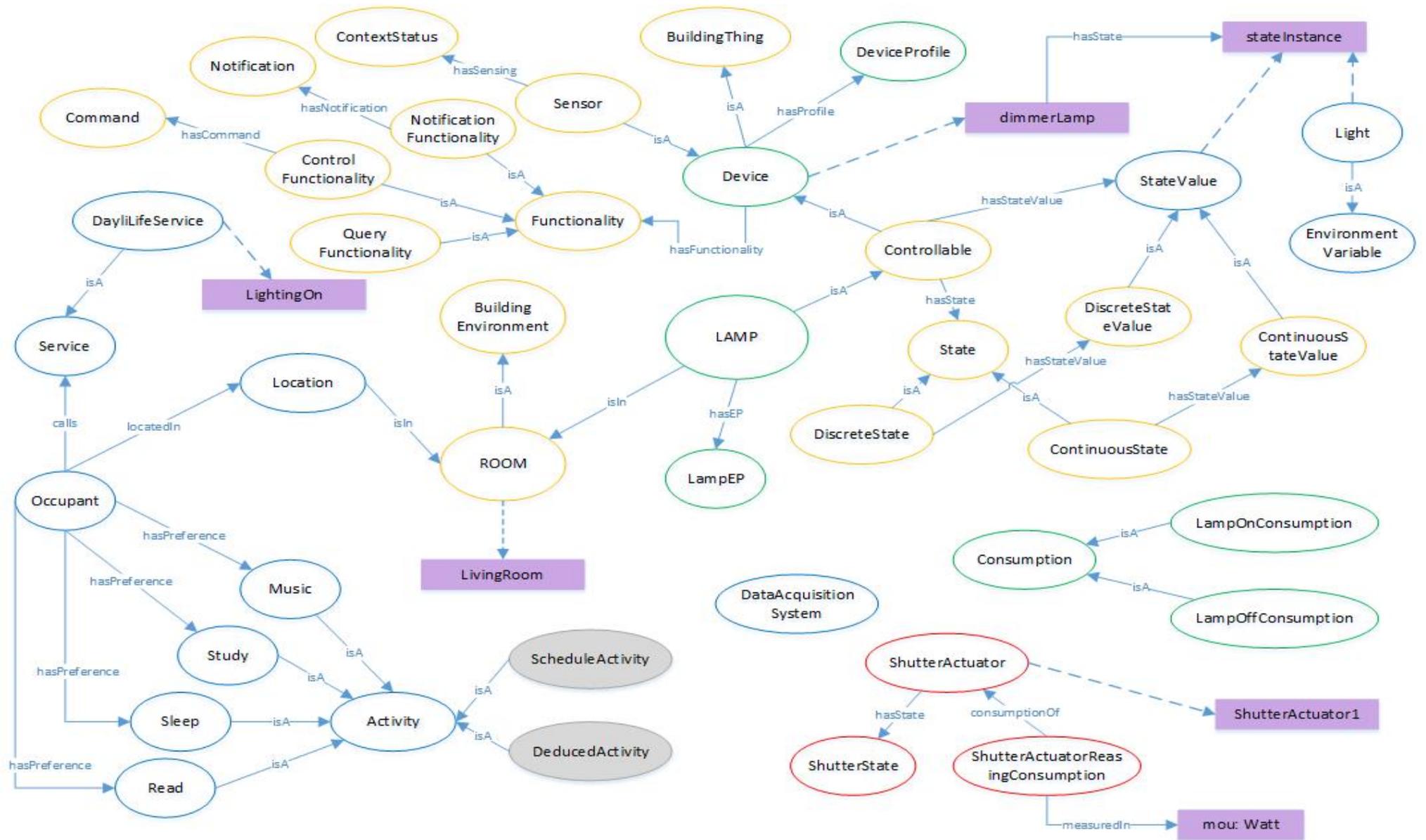


Ilustración 15: Diagrama RDF

Tabla 1: Glosario de Términos

<i>Término</i>	<i>Acrónimo</i>	<i>Descripción</i>	<i>Tipo</i>
<i>BuildingEnvironment</i>	BE	Espacios delimitados de una residencia.	Concepto
<i>Light</i>		Variable del entorno, iluminación.	Concepto
<i>EnvironmentVariable</i>	EV	Variable den entorno, como la iluminación, el calor o el ruido.	Concepto
<i>Location</i>			Concepto
<i>Room</i>		Habitación del departamento.	Concepto
<i>Occupant</i>			Concepto
<i>Activity</i>		Actividad de una persona que activa un escenario.	Concepto
<i>DataAcquisitionSystem</i>			Concepto
<i>DeduceActivity</i>			Concepto
<i>ShutterActuator</i>	SA		Concepto
<i>Consumption</i>		El consumo del dispositivo..	Concepto
<i>StateValue</i>	SV		Concepto
<i>State</i>		Estado de un dispositivo actuador	Concepto
<i>Sensor</i>			Concepto
<i>LamEP</i>	LEP	Perfil de la energía de la lámpara.	Concepto
<i>Notification</i>			Concepto
<i>Functionality</i>			Concepto
<i>Command</i>			Concepto



4.2. Relaciones binarias de una Ontología con dispositivos de un departamento domotizado y patrón de comportamiento de personas habitantes en dicho departamento.

Tabla 2: Tabla de Relaciones Binaria

Nombre de Relación	Concepto Fuente	Concepto Destino	Cardinalidad	Relación Inversa
isIlluminatedByMediumArtificialIllumination	Location	MediumArtificialIllumination	1:1	mediumArtificialIlluminationIlluminates
isIlluminatedByNoArtificialIllumination	Location	NotArtificialIllumination	1:1	noArtificialIlluminationIlluminates
noArtificialIlluminationIlluminates	NotArtificialIllumination	Location	1:1	isIlluminatedByNoArtificialIllumination
isIlluminatedByHighArtificialIllumination	Location	HighArtificialIllumination	1:1	highArtificialIlluminationIlluminates
mediumArtificialIlluminationIlluminates	MediumArtificialIllumination	Location	1:1	isIlluminatedByMediumArtificialIllumination
highArtificialIlluminationIlluminates	HighArtificialIllumination	Location	1:1	isIlluminatedByHighArtificialIllumination
calls	Occupant	Service	1:n	isCalledBy
isCalledBy	Service	Occupant	n:1	calls
consumptionOf	Consumption	Device	1:1	
hasConsumption	Device	Consumption	1:1	
hasCommand	ControlFunctionality	Command	1:n	
hasEP	Device	Profile	1:1	

hasSimpleLampEP	SimpleLamp	SimpleLampEP	1:1	
hasDimmerLampEP	DimmerLamp	DimmerLampEP	1:1	
hasFunctionality	Device	Functionality	1:n	
hasLamps	Location	Lamp	1:n	
hasLight	Activity	ArtificialLight	1:1	
necessaryLamps	ArtificialLight	Lamp	1:1	
hasNaturalLightCondition	ArtificialLight	NaturalLight	1:1	
sensingNaturalLight	NaturalLightSensor	NaturalLight	1:1	
hasNotification	NotificationFunctionality	Notification	1:n	
hasOccupant	Location	Occupant	1:1	locatedIn
hasPreference	Occupant	Activity	1:n	
hasSensing	Sensor	ContextStatus	1:n	
hasState	Controllable	State	1:1	
isIn	Location	room	1:1	
locateIn	Occupant	Location	1:1	hasOccupant
hasPrecondition	State	Activity	1:n	

Tabla 3: Descripción de Relaciones Binarias

Relaciones	Descripción
isIlluminatedByMediumArtificialIllumination	Relación entre una locación (Location) y el nivel de intensificación de la luz artificial (MediumArtificialIllumination), es asimétrica e irreflexiva.
isIlluminatedByNoArtificialIllumination	Relación entre una locación (Location) y el nivel de intensificación de la luz artificial (NotArtificialIllumination), es asimétrica e irreflexiva.

noArtificialIlluminationIlluminates	Relación entre el nivel de intensificación de la luz artificial (NotArtificialIlluminaton) y una locación (Location), es asimétrica e irreflexiva.
isIlluminatedByHighArtificialIllumination	Relación entre una locación (Location) y el nivel de intensificación de la luz artificial (HighArtificialIlluminaton), es asimétrica e irreflexiva.
mediumArtificialIlluminationIlluminates	Relación entre el nivel de intensificación de la luz artificial (MediumArtificialIlluminaton) y una locación (Location), es asimétrica e irreflexiva.
highArtificialIlluminationIlluminates	Relación entre el nivel de intensificación de la luz artificial (HighArtificialIlluminaton) y una locación (Location), es asimétrica e irreflexiva.
calls	Relación entre un ocupante (Occupant) y los servicios (Service) que solicita, es asimétrica e irreflexiva.
isCalledBy	Relación entre un servicio (Service) y el ocupante (Occupant) que solicita dicho servicio, es asimétrica e irreflexiva.
consumptionOf	Relación entre el consumo (Consumption) y un dispositivo (Device), no funcional, no transitiva, asimétrica e irreflexiva.
hasConsumption	Relación entre un dispositivo (Device) y su consumo (Consumption), no inversa, no transitiva, asimétrica e irreflexiva.
hasCommand	Relación entre la funcionalidad de control y sus comandos (Command), no inversa, no funcional, no transitiva, asimétrica e irreflexiva.
hasEP	Relación entre la lámpara (Lamp) y su perfil de consumo de energía, funcional, no transitiva, asimétrica e irreflexiva.
hasSimpleLampEP	Relación entre una lámpara simple y su perfil de consumo de energía específico, funcional, no transitiva, asimétrica e irreflexiva.
hasDimmerLampEP	Relación entre entre una lámpara regulable (Dimmer) y su perfil de consumo de energía específico, funcional, no transitiva, asimétrica e irreflexiva.
hasFunctionality	Relación entre el dispositivo (Device) y su funcionalidad (Functionality), no inversa, asimétrica e irreflexiva.

hasLamps	Relación entre una ubicación y las lámparas que en ella se encuentran, la cantidad de las lámparas, su tipo y su estado determinan la iluminación artificial de dicha ubicación, no inversa, no funcional, no transitiva, asimétrica e irreflexiva.
hasLight	Relación entre una actividad (Activity) y el nivel de luz artificial que necesita para ser llevada a cabo sin contratiempo. Funcional, no transitiva, asimétrica e irreflexiva.
necessaryLamps	Relación entre la luz artificial y las lámparas (Lamp) necesarias para generarla, las cuales según la regulación de su intensidad determinan el nivel de luz artificial. Es asimétrica e irreflexiva.
hasNaturalLightCondition	Relación entre la luz artificial y la luz natural, ya que la primera depende de la intensidad de la segunda. Es funcional, asimétrica e irreflexiva.
sensingNaturalLight	Indica la relación entre el sensor de luz natural y la luz natural del ambiente. Es asimétrica e irreflexiva.
hasNotification	Relación entre la funcionalidad de notificación y la notificación generada, es asimétrica e irreflexiva.
hasOccupant	Indica la ubicación y el ocupante que se encuentra en dicha ubicación, es asimétrica e irreflexiva.
hasPreference	Relación entre el ocupante y las actividades de su preferencia, es asimétrica e irreflexiva.
hasState	Relación entre el dispositivo controlable y su estado, es asimétrica e irreflexiva.
isIn	Relación que indica una ubicación en una habitación, es asimétrica e irreflexiva.
locatedIn	Relación entre el ocupante y la ubicación donde se encuentra, es asimétrica e irreflexiva.

---

4.3. Atributos, axiomas y reglas de inferencia de una Ontología con dispositivos de un departamento domotizado y patrón de comportamiento de personas habitantes en dicho departamento.

## ATRIBUTOS

En el software Protégé los atributos son conocidos como DataProperty.

Tabla 4: Atributos

<b>DataProperty</b>	<b>Dominio</b>	<b>Rango</b>
facePattern	Occupant	String
measuredIn	ShutterActuatorReasingConsumption	Float
initialTimeInMinutes	Activity	Integer
finalTimeInMinutes	Activity	Integer
weekDay	Activity	String
naturalIlluminationLevel	NaturalLight	Float
hasContinuousStateValue	ContinuousState	Float
hasDiscreteStateValue	DiscreteState	boolean
hasNotificarionValue	Notification	String
dimension	BuildingEnviroment, Location	Float
identifier	Occupant, BuildingThing, Service, Command, Functionality, Notification, Activity	Integer
Parameter	Command	Integer, Float
meanConsumtion (Consumo promedio)	DeviceProfile	Float
age	Occupant	Integer
timeOfWakeUp	Occupant	dateTime

timeOfSleep	Occupant	dateTime
leaveTime	Occupant	dateTime
arrivalTime	Occupant	dateTime

---

## AXIOMAS

En las tablas 5 y 6 se presentan fragmentos de las tablas de definición de condiciones necesarias y suficientes de la ontología, respectivamente.

**Tabla 5: Condiciones necesarias de algunas clases de la ontología**

N°	Clase	Condiciones necesarias(subclases)
1	MusicAtNight	hasLight only MediumArtificialIllumination Music
2	ReadAtNight	hasLight only HighArtificialIllumination Read
3	VeryArtificiallyIlluminatedLocation	hasLamps min 1 HighIntensityDimmerLamp hasLamps min 1 OnSimpleLamp hasLamps only ((hasLamps min 1 OnDimmerLamp) or (hasLamps min 1 OnSimpleLamp)) Location
4	DimmerLamp	HasDimmerLampEP exactly 1 DimmerLampEP Lamp

Tabla 6: Condiciones suficientes de algunas clases de la ontología

N°	Clase	Condiciones suficientes (clases equivalentes)
1	MusicAtNight	Music and (initialTimeInMinutes value 1140)
2	ReadAtNight	Read and (initialTimeInMinutes value 1140)
3	VeryArtificiallyIlluminatedLocation	-
4	DimmerLamp	Lamp and (hasState only ContinuousState)

## REGLAS DE INFERENCIA

Para describir las reglas se sigue la forma:

Si <condiciones> entonces <consecuencias o acciones>

- Si Occupant is MusicAtNight Entonces Light MediumArtificialIllumination.

Si un ocupante con una preferencia de escuchar música de noche llega a una localización, esta última se verá iluminada a nivel medio.

- Si Occupant is ReadAtNight Entonces Light HighArtificialIllumination.

Si un ocupante con una preferencia de leer de noche llega a una localización, esta última se verá iluminada a nivel alto.

- Si Occupant is StudyAtNight Entonces Light HighArtificialIllumination.

Si un ocupante con una preferencia de estudiar de noche llega a una localización, esta última se verá iluminada a nivel alto.

- Si Occupant is Sleep Entonces Light NotArtificiallyIlluminatedLocation.

Si un ocupante con una preferencia de dormir se encuentra en una localización determinada, esta última no se verá iluminada.

4.4. Implementar la Ontología anteriormente desarrollada utilizando el software Protégé y ejecutarla mediante el razonador Jena 2.1 en un entorno Web.

#### IMPLEMENTACIÓN DE LA ONTOLOGÍA DESARROLLADA EN PROTÉGÉ

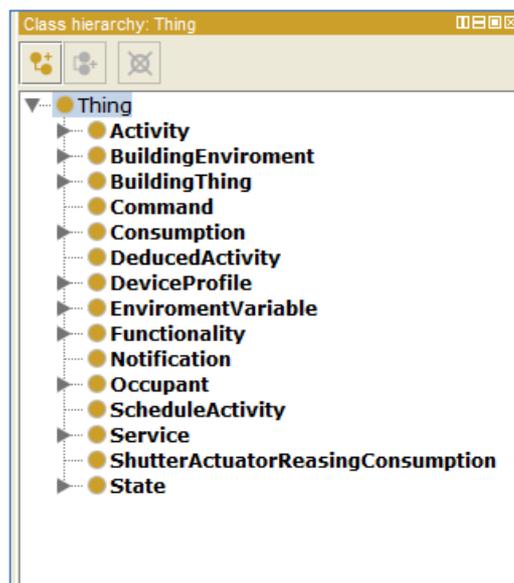


Ilustración 17: Jerarquía completa de clases de la ontología

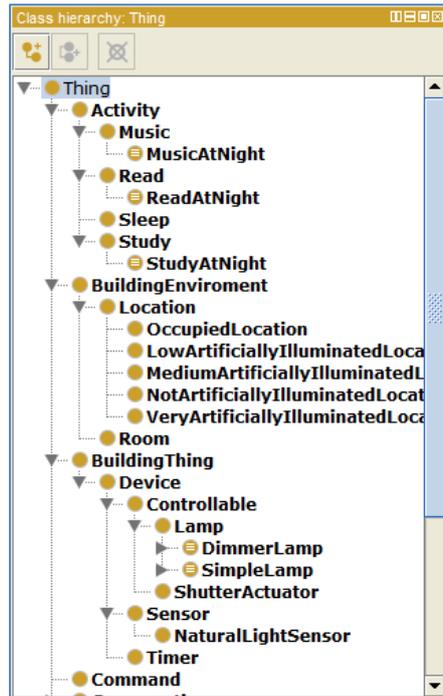


Ilustración 18: Jerarquía desglosada de clases de la ontología

En Protégé las propiedades representan los atributos de las clases y sus instancias y también cuenta con restricciones.

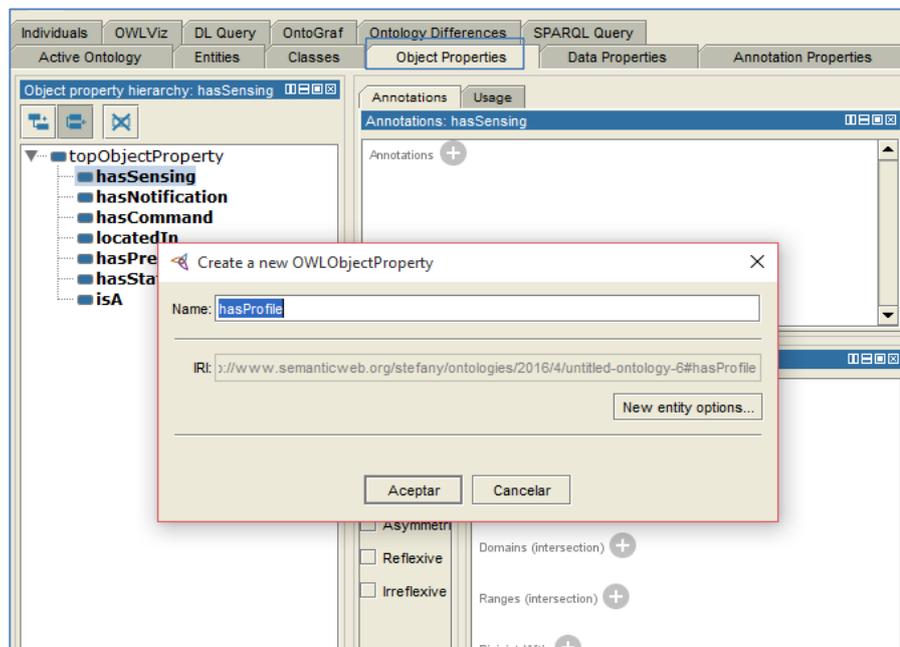


Ilustración 19: Creación de Object Properties

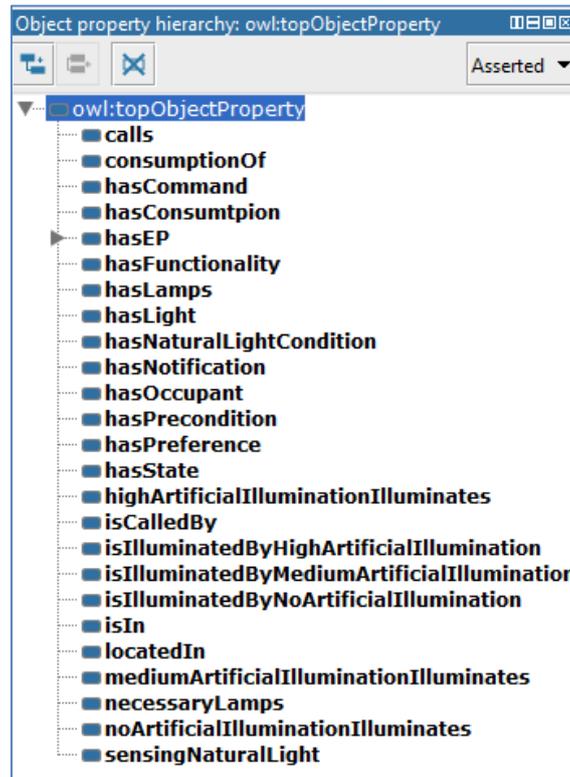


Ilustración 20: Jerarquía de propiedades de objetos

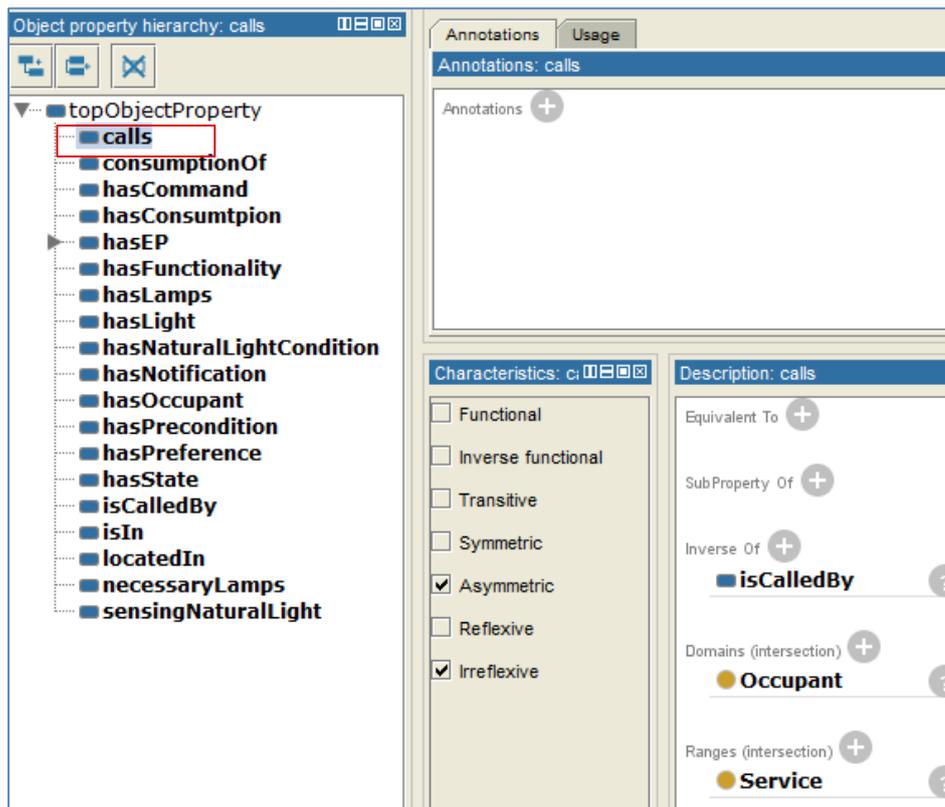


Ilustración 21: Característica property calls

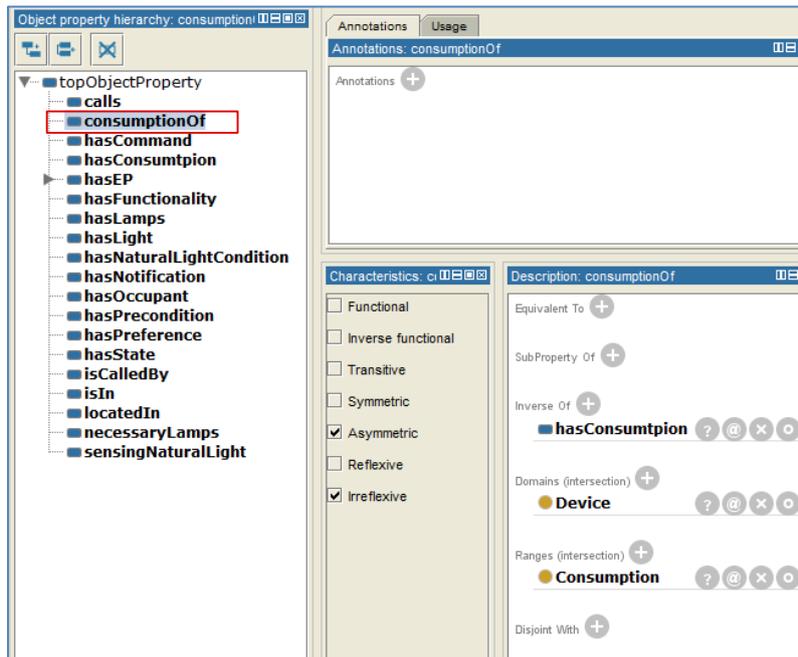


Ilustración 22: Característica property consumptionOf

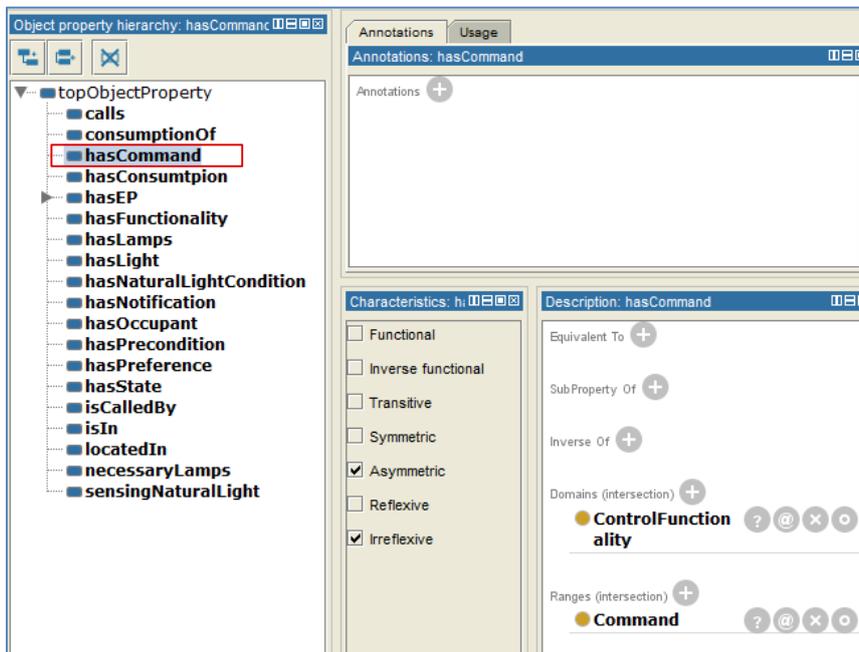


Ilustración 23: Característica property hasCommand

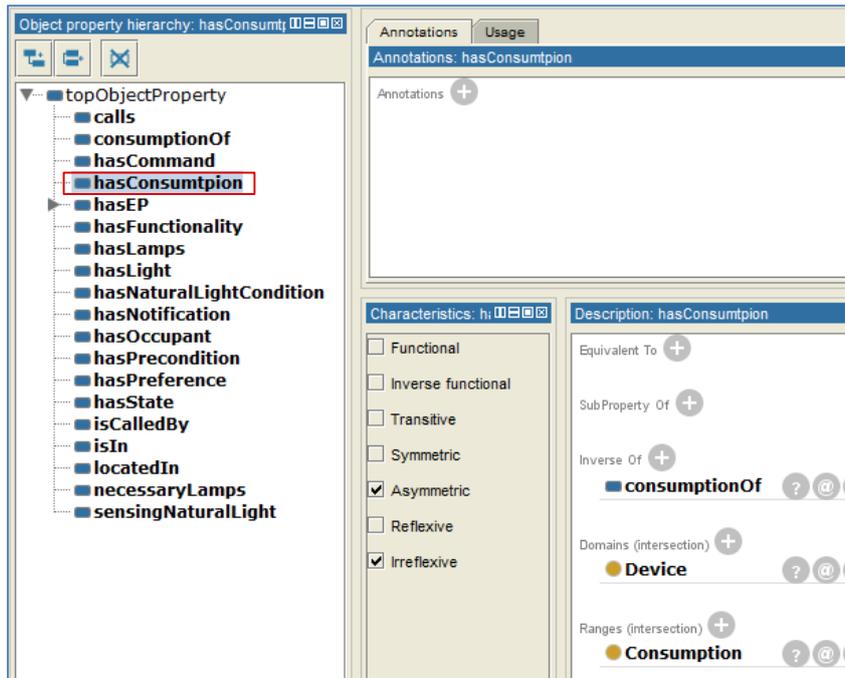


Ilustración 24: Característica property hasConsumption

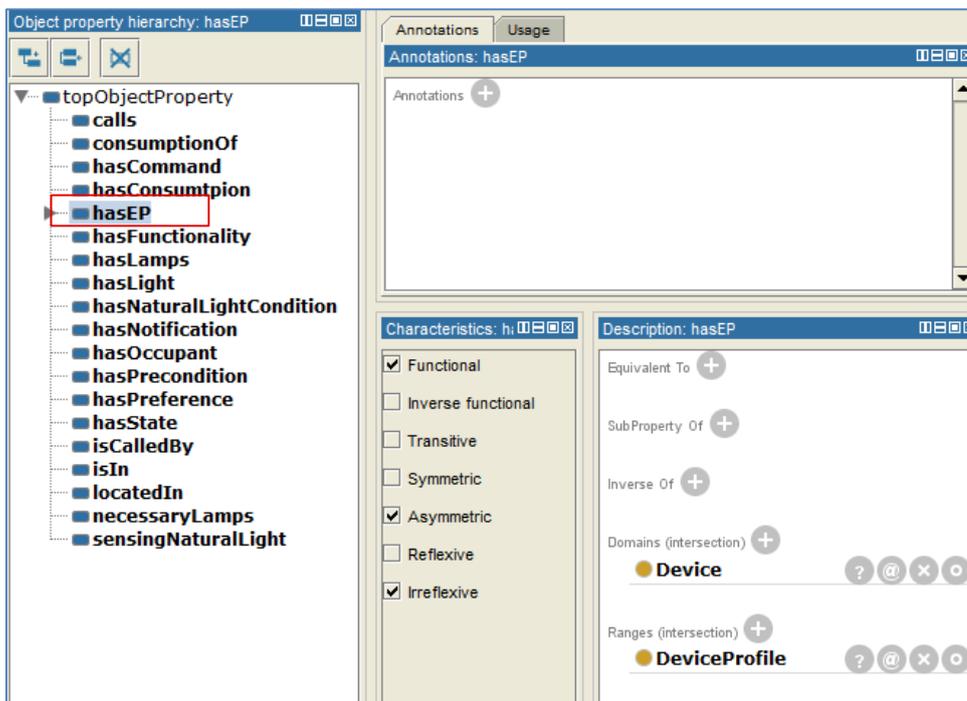


Ilustración 25: Característica property hasEP

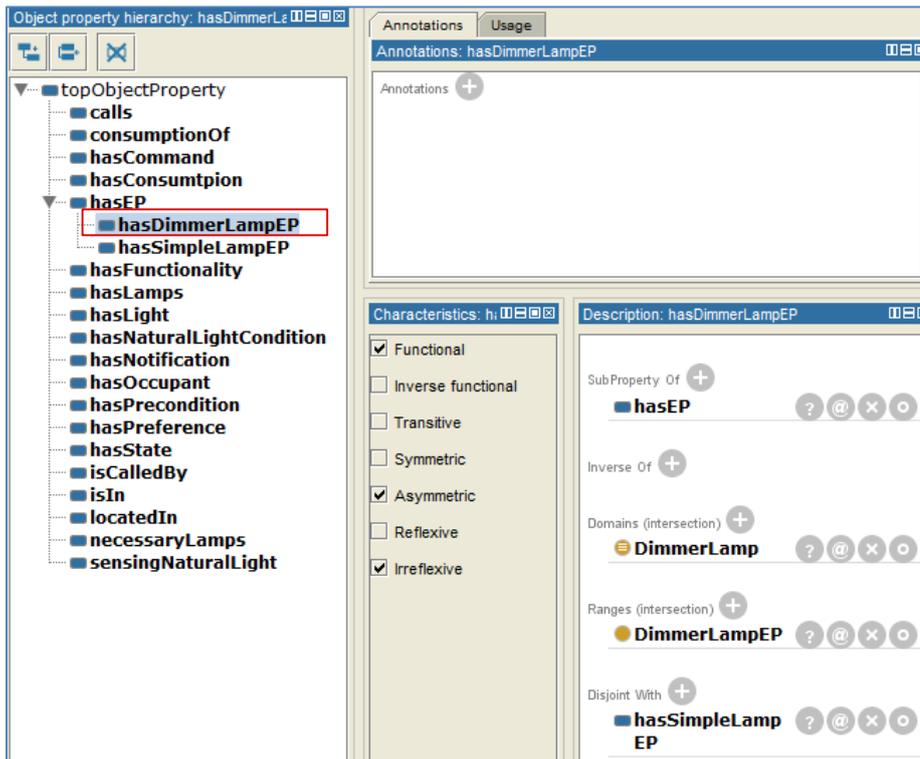


Ilustración 26: Característica property hasDimmerLampEP

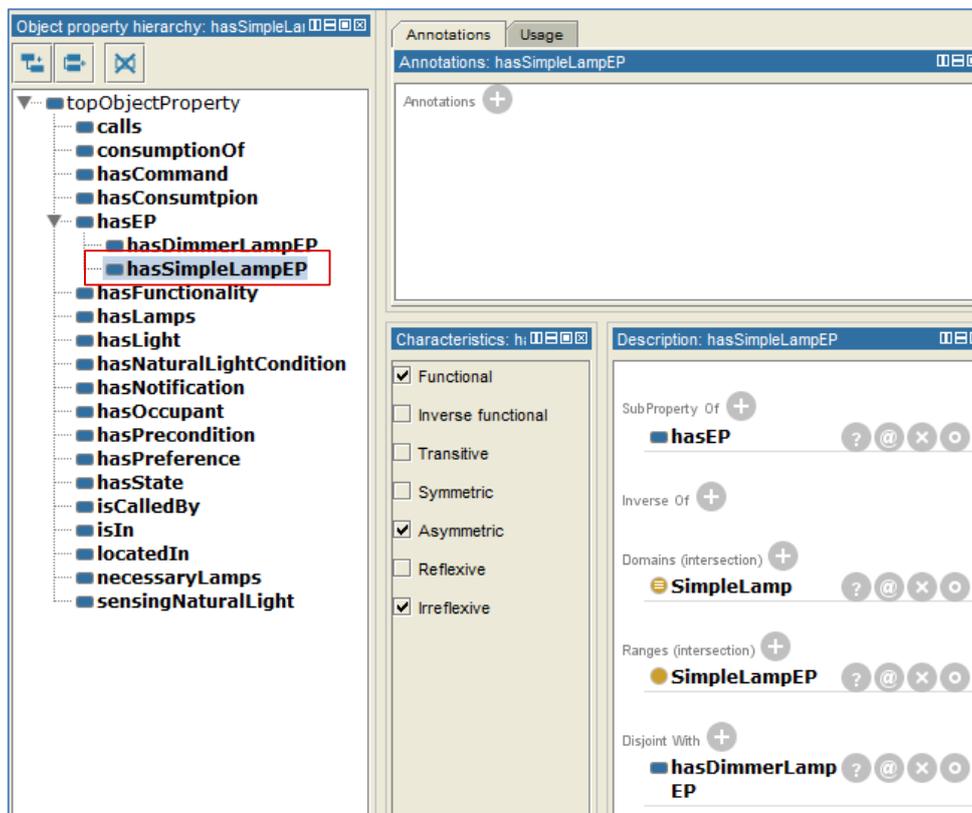


Ilustración 27: Característica property hasSimpleLampEP

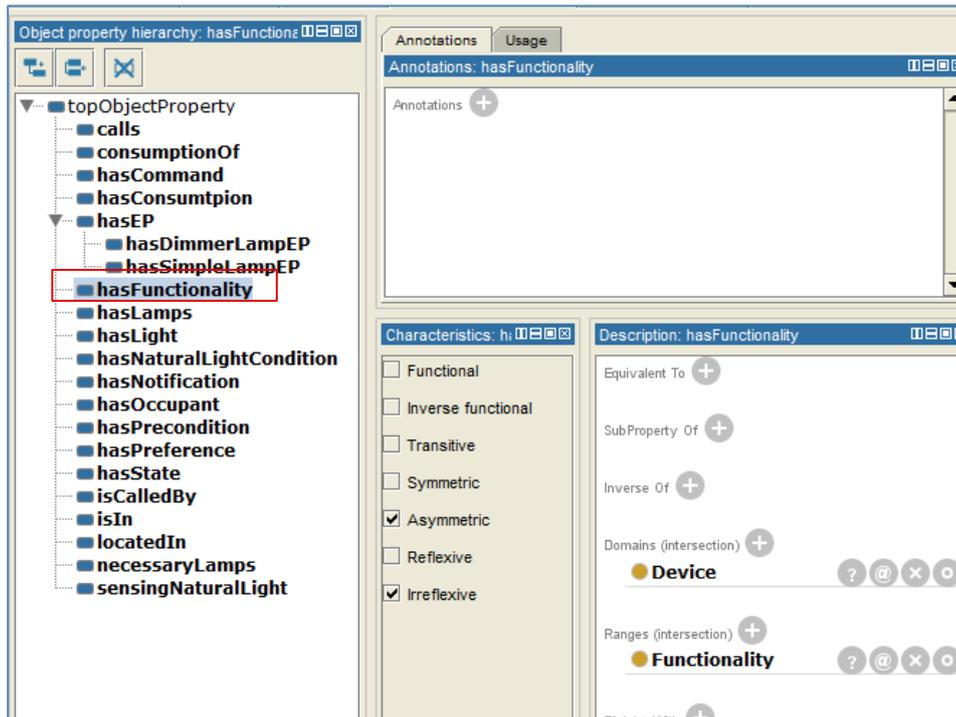


Ilustración 28: Característica property hasFunctionality

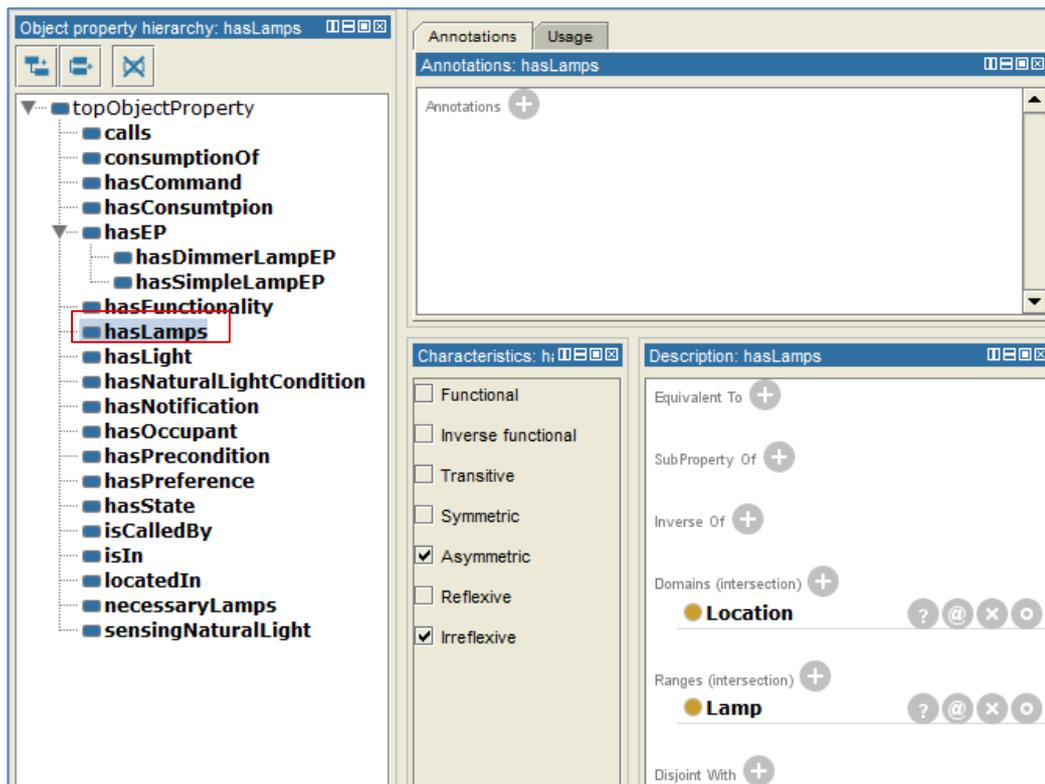


Ilustración 29: Característica property hasLamps

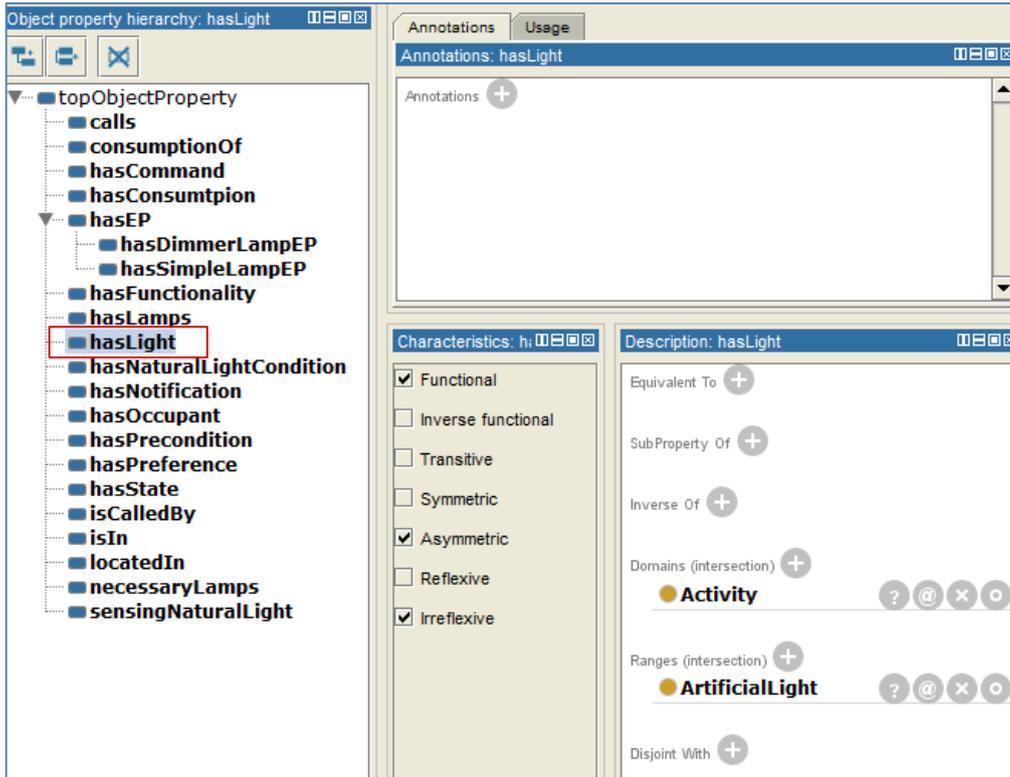


Ilustración 30: Característica property hasLight

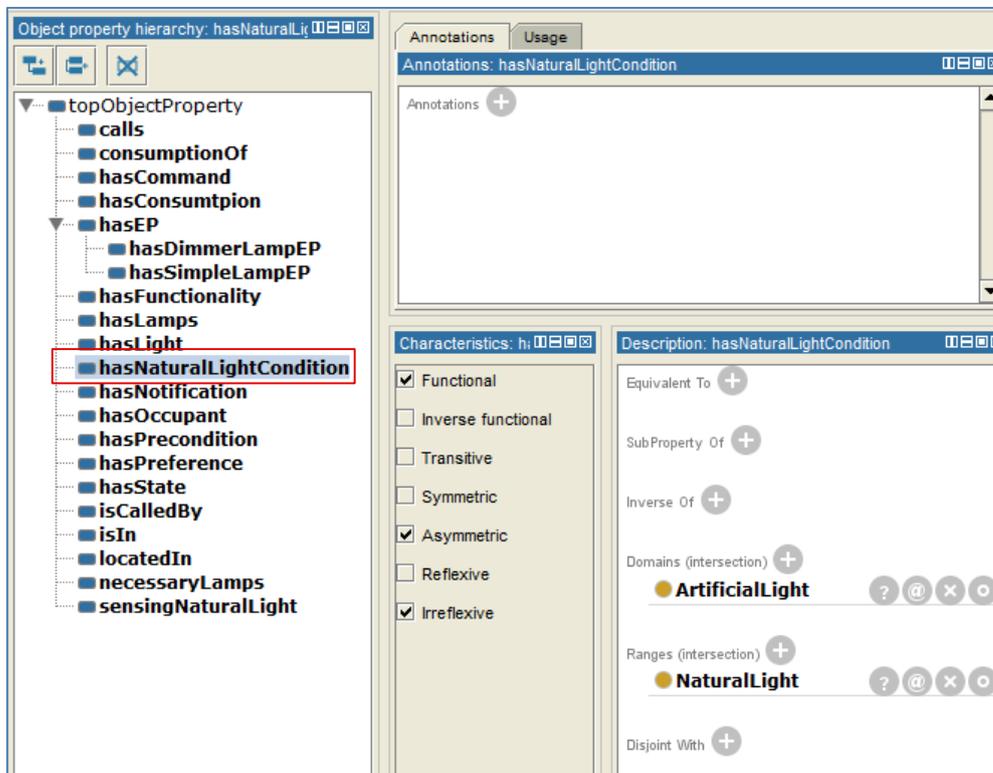


Ilustración 31: Característica property hasNaturalLightCondition

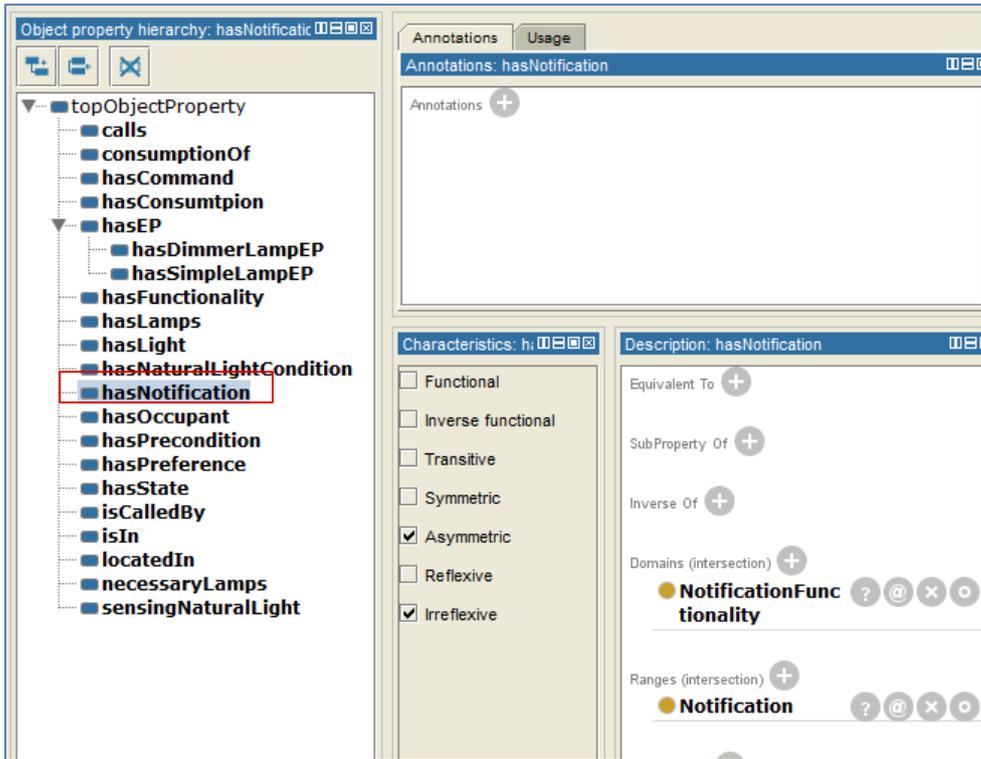


Ilustración 32: Característica property has Notification

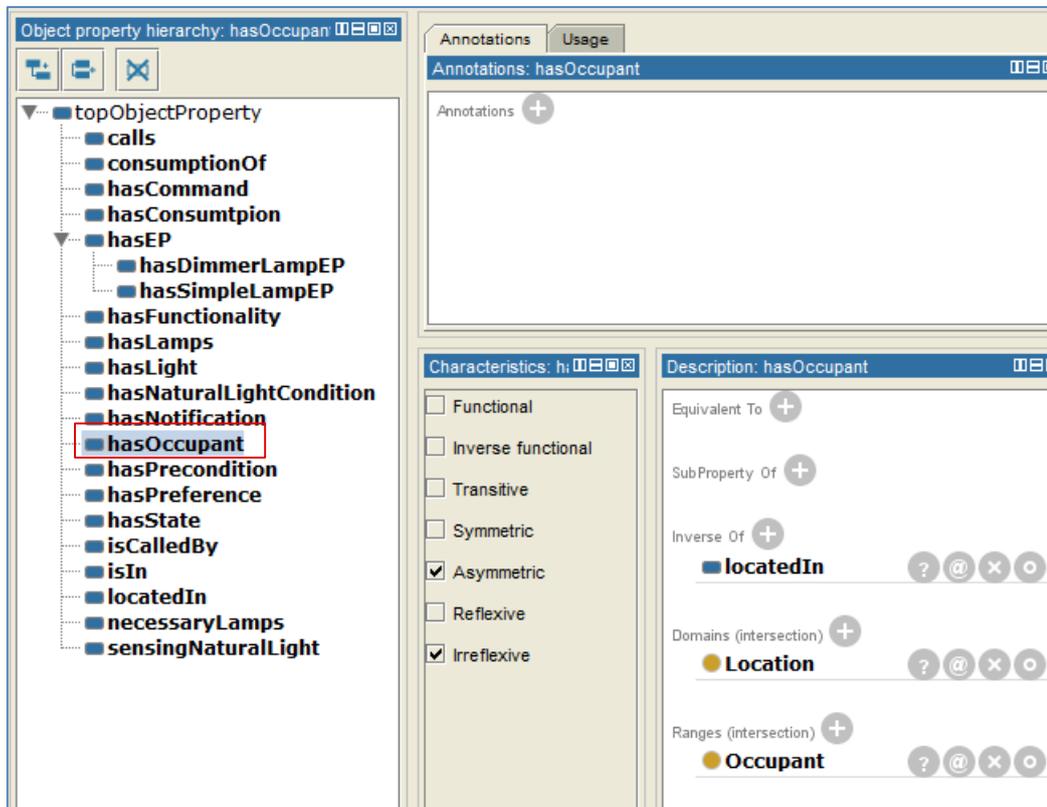


Ilustración 33: Característica property hasOccupant

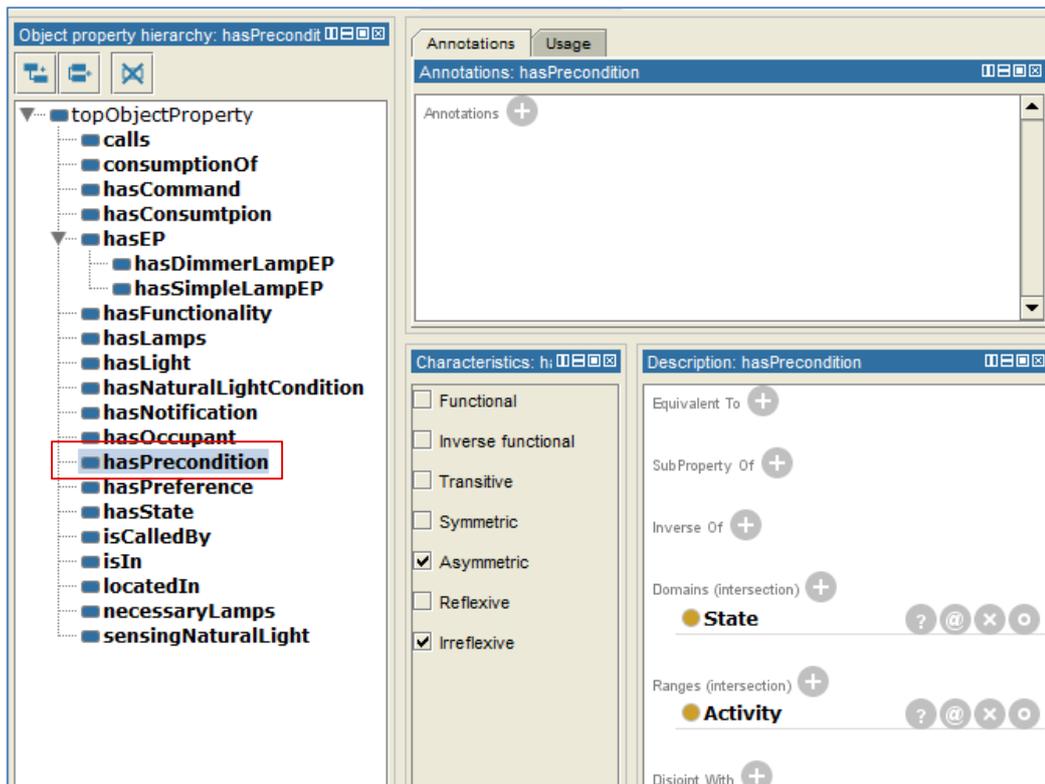


Ilustración 34: Característica property hasPrecondition

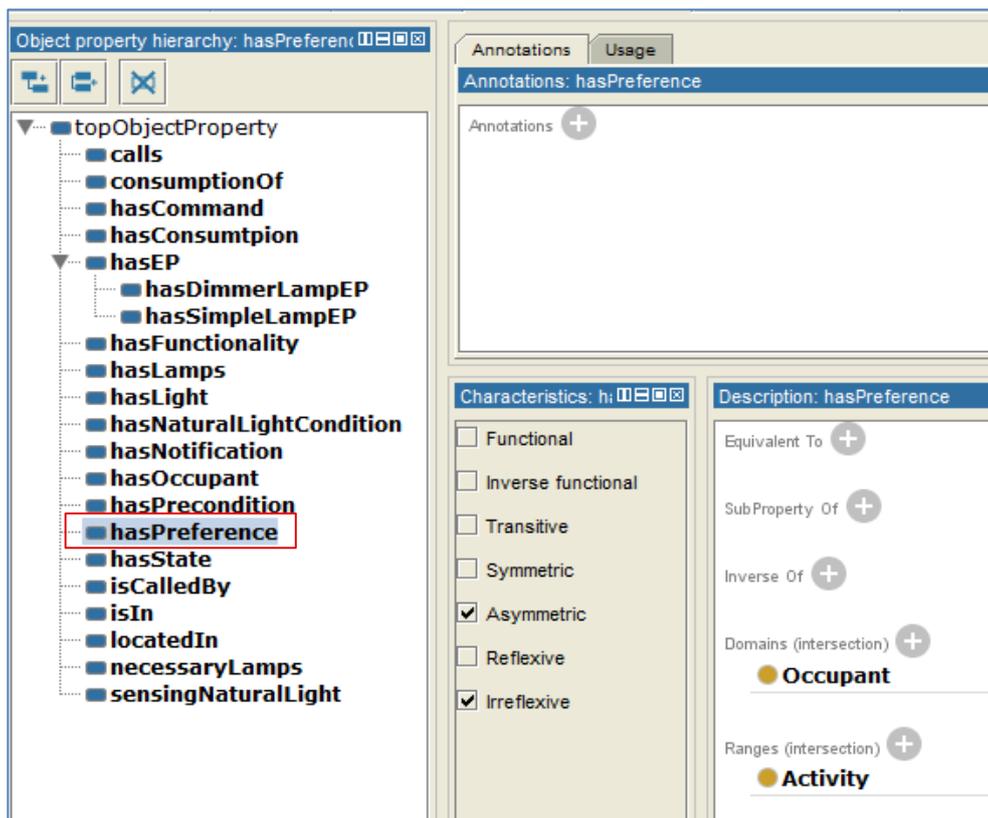


Ilustración 35: Característica property hasPreference

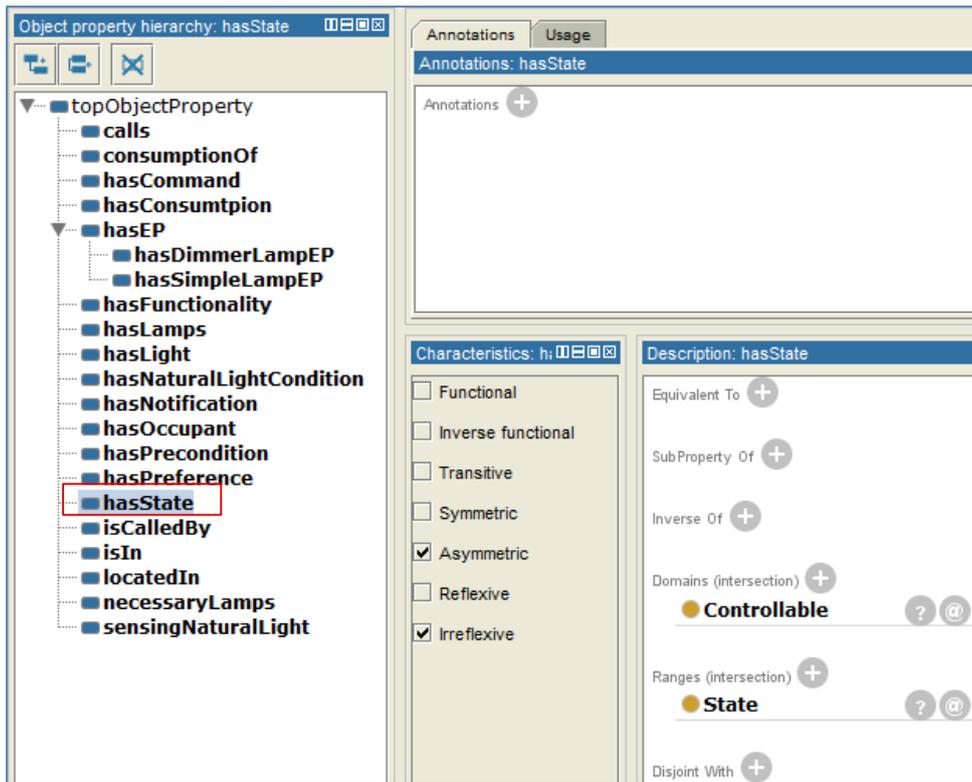


Ilustración 36: Característica property hasState

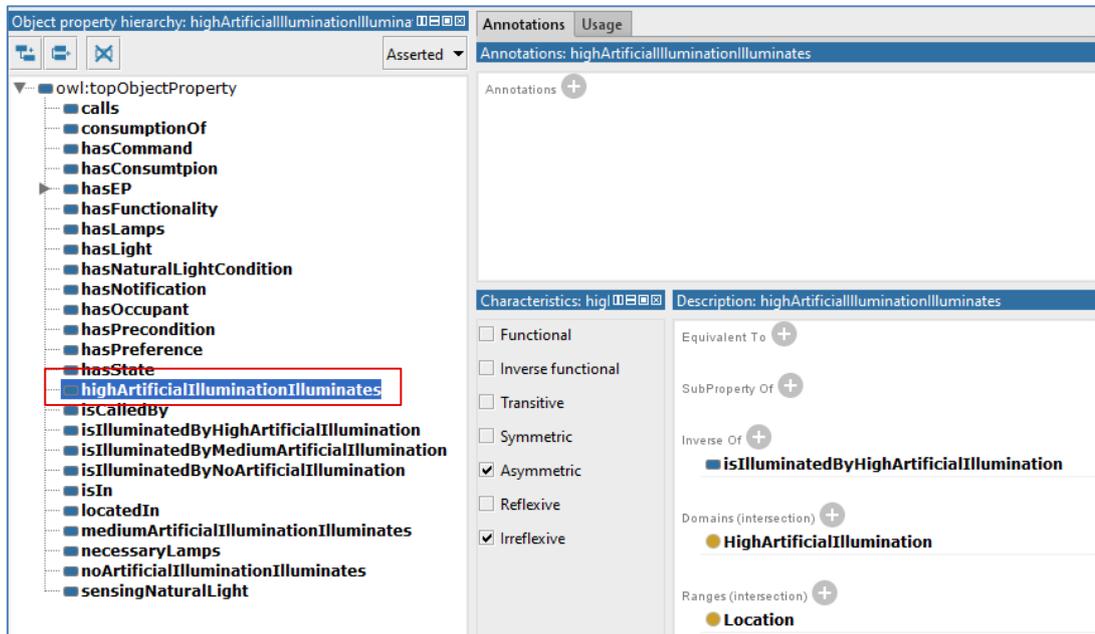


Ilustración 37: Característica property highArtificialIlluminationIlluminates

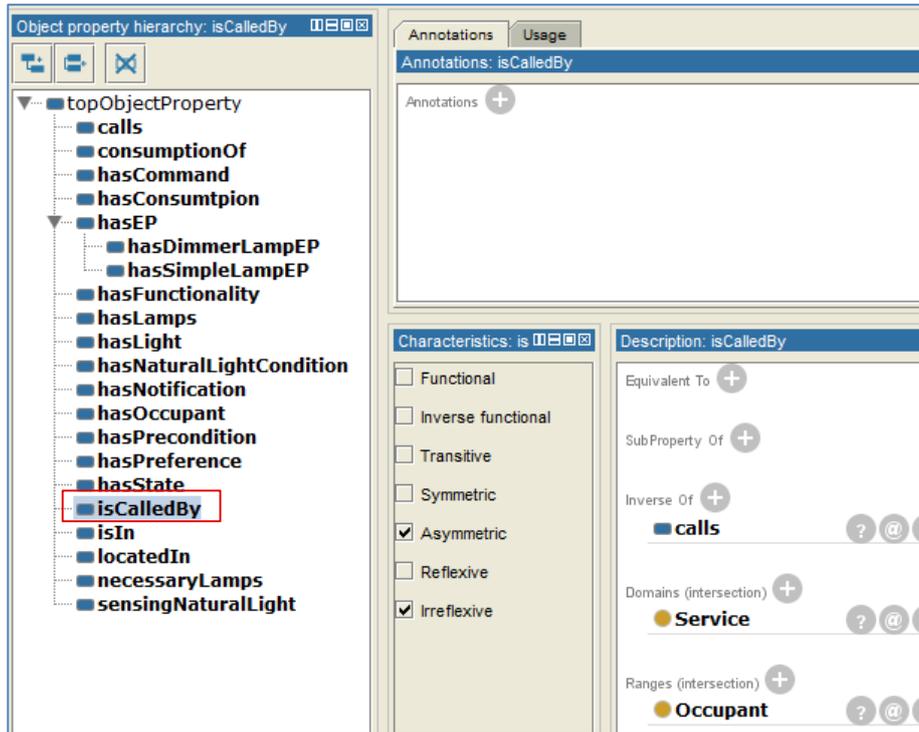


Ilustración 38: Característica property isCalledBy

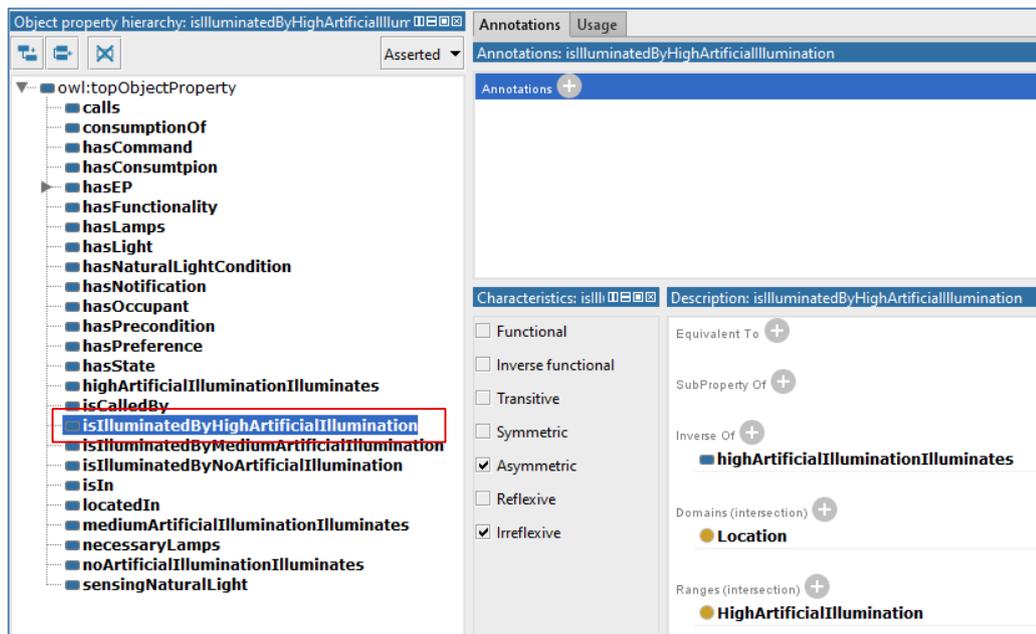


Ilustración 39: Característica property isIlluminatedByHighArtificialIllumination

Object property hierarchy: isIlluminatedByMediumArtificialIllumination

- owl:topObjectProperty
  - calls
  - consumptionOf
  - hasCommand
  - hasConsumption
  - hasEP
  - hasFunctionality
  - hasLamps
  - hasLight
  - hasNaturalLightCondition
  - hasNotification
  - hasOccupant
  - hasPrecondition
  - hasPreference
  - hasState
  - highArtificialIlluminationIlluminates
  - isCalledBy
  - isIlluminatedByHighArtificialIllumination
  - isIlluminatedByMediumArtificialIllumination**
  - isIlluminatedByNoArtificialIllumination
  - isIn
  - locatedIn
  - mediumArtificialIlluminationIlluminates
  - necessaryLamps
  - noArtificialIlluminationIlluminates
  - sensingNaturalLight

Annotations: isIlluminatedByMediumArtificialIllumination

Annotations +

Characteristics: isIlluminatedByMediumArtificialIllumination

- Functional
- Inverse functional
- Transitive
- Symmetric
- Asymmetric
- Reflexive
- Irreflexive

Description: isIlluminatedByMediumArtificialIllumination

Equivalent To +

SubProperty Of +

Inverse Of +

- mediumArtificialIlluminationIlluminates

Domains (intersection) +

- Location

Ranges (intersection) +

- MediumArtificialIllumination

Ilustración 40: Característica property isIlluminatedByMediumArtificialIllumination

Object property hierarchy: isIlluminatedByNoArtificialIllumination

- owl:topObjectProperty
  - calls
  - consumptionOf
  - hasCommand
  - hasConsumption
  - hasEP
  - hasFunctionality
  - hasLamps
  - hasLight
  - hasNaturalLightCondition
  - hasNotification
  - hasOccupant
  - hasPrecondition
  - hasPreference
  - hasState
  - highArtificialIlluminationIlluminates
  - isCalledBy
  - isIlluminatedByHighArtificialIllumination
  - isIlluminatedByMediumArtificialIllumination
  - isIlluminatedByNoArtificialIllumination**
  - isIn
  - locatedIn
  - mediumArtificialIlluminationIlluminates
  - necessaryLamps
  - noArtificialIlluminationIlluminates
  - sensingNaturalLight

Annotations: isIlluminatedByNoArtificialIllumination

Annotations +

Characteristics: isIlluminatedByNoArtificialIllumination

- Functional
- Inverse functional
- Transitive
- Symmetric
- Asymmetric
- Reflexive
- Irreflexive

Description: isIlluminatedByNoArtificialIllumination

Equivalent To +

SubProperty Of +

Inverse Of +

- noArtificialIlluminationIlluminates

Domains (intersection) +

- Location

Ranges (intersection) +

- NotArtificialIllumination

Ilustración 41: Característica property isIlluminatedByNoArtificialIllumination

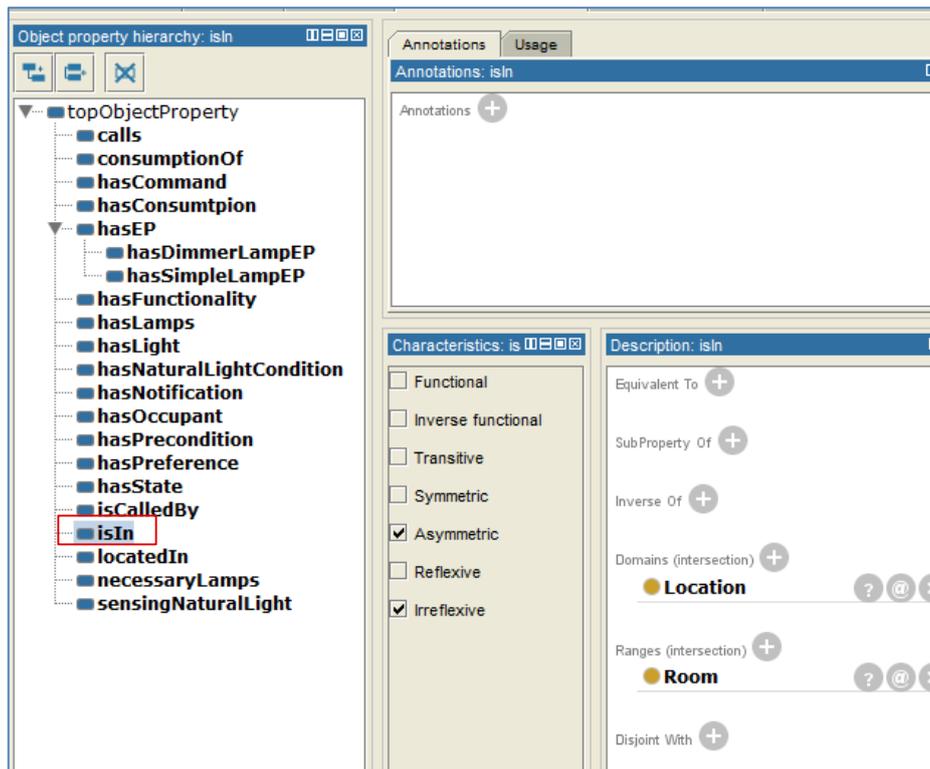


Ilustración 42: Característica property isln

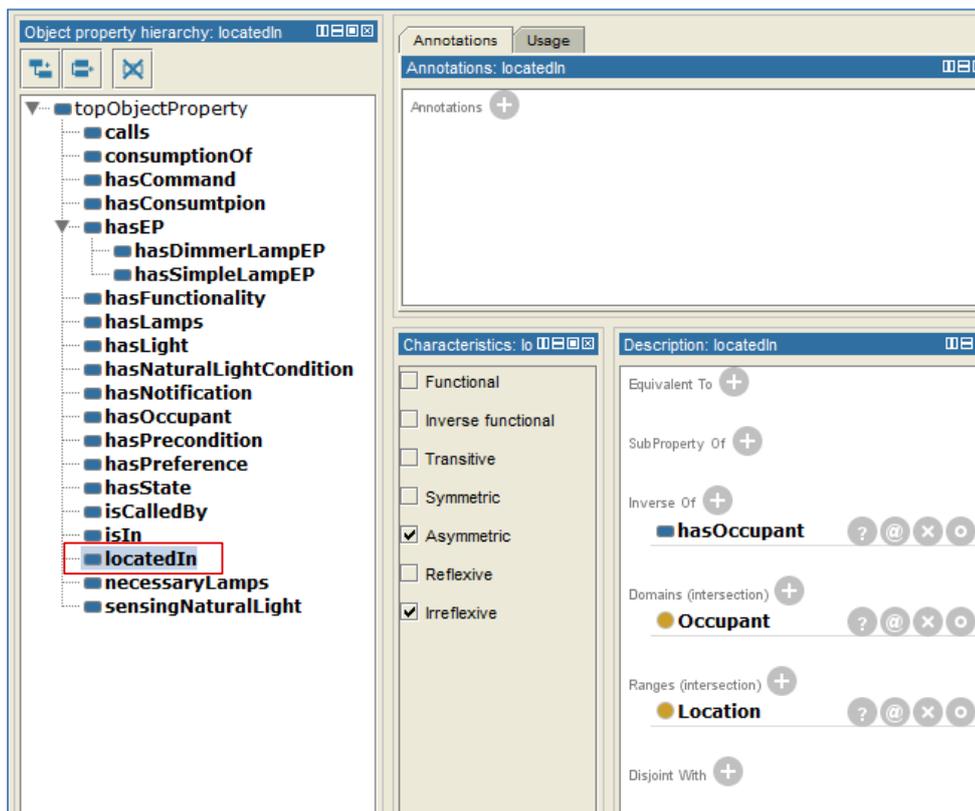


Ilustración 43: Característica property locatedIn

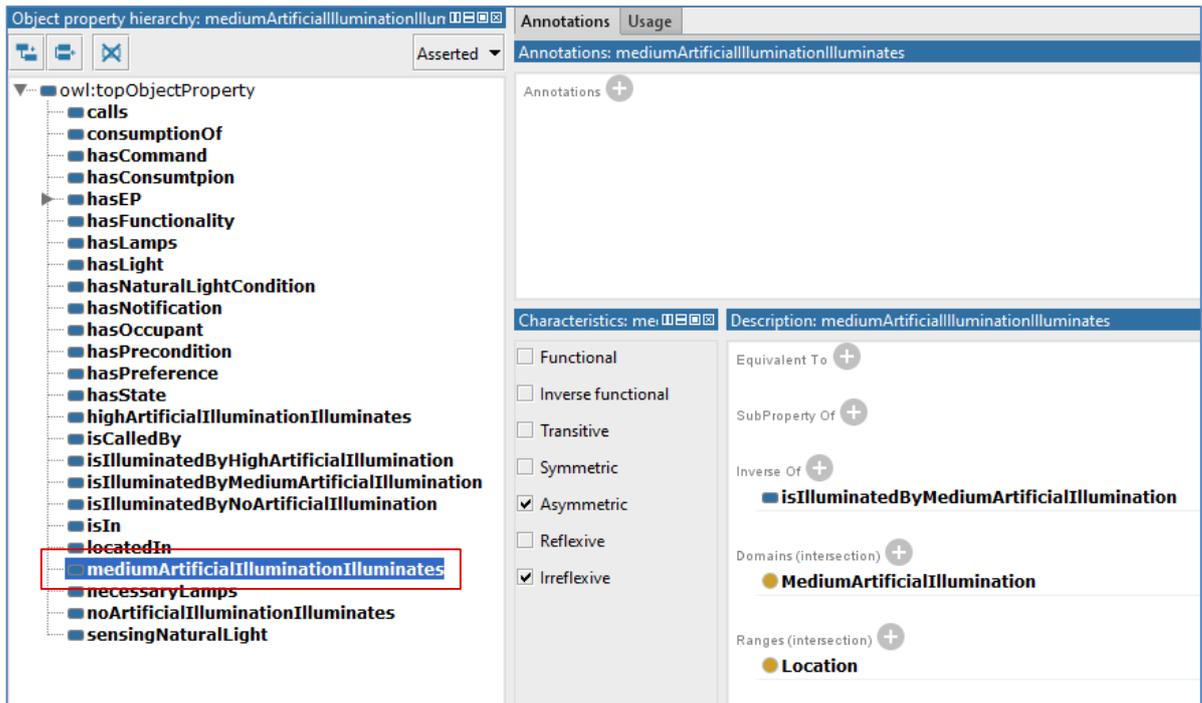


Ilustración 44: Característica property mediumArtificialIlluminationIlluminates

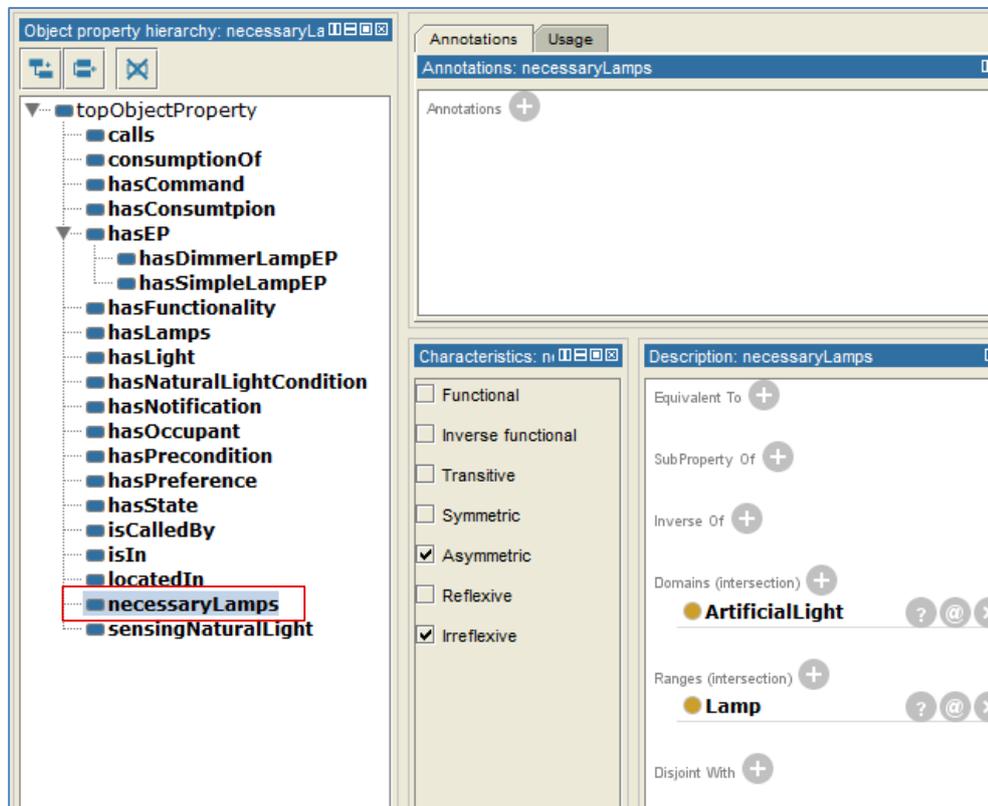


Ilustración 45: Característica property necessaryLamps

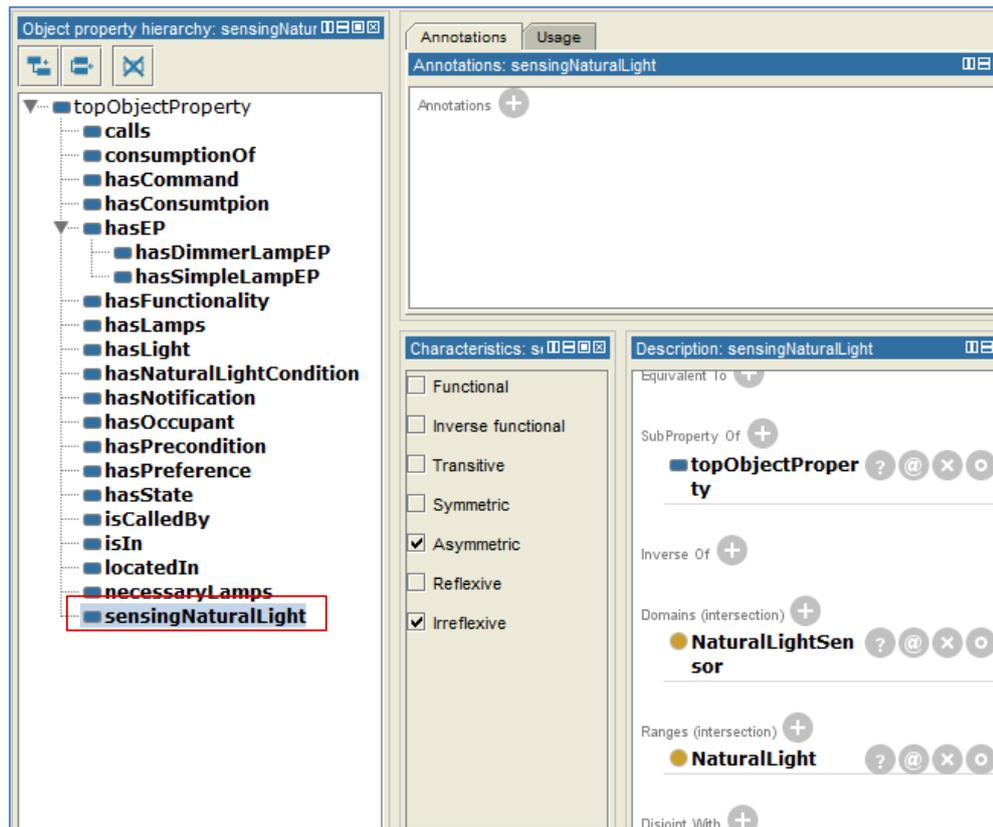


Ilustración 46: Características property sensingNaturalLight

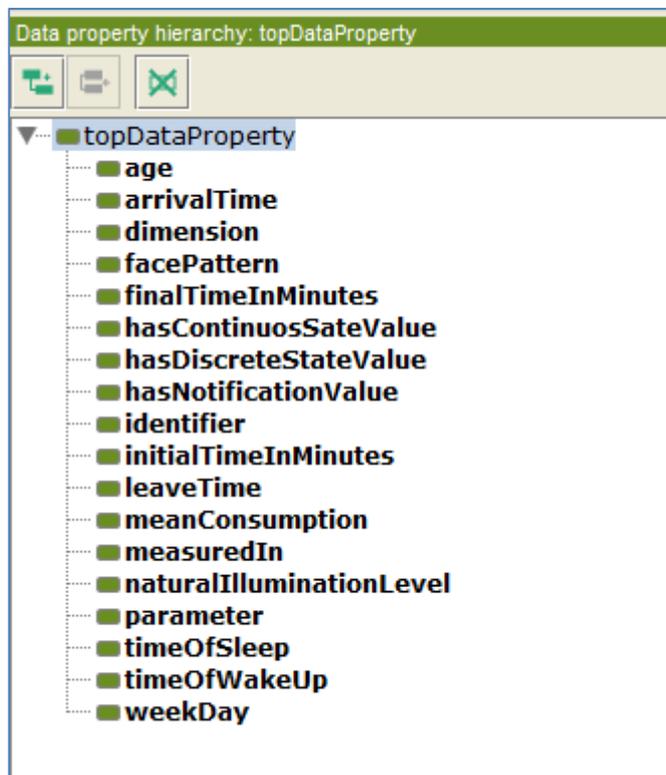


Ilustración 47: Jerarquía de datos de la ontología en Protegé

## CREACIÓN DE LOS AXIOMAS PARA LA ONTOLOGÍA

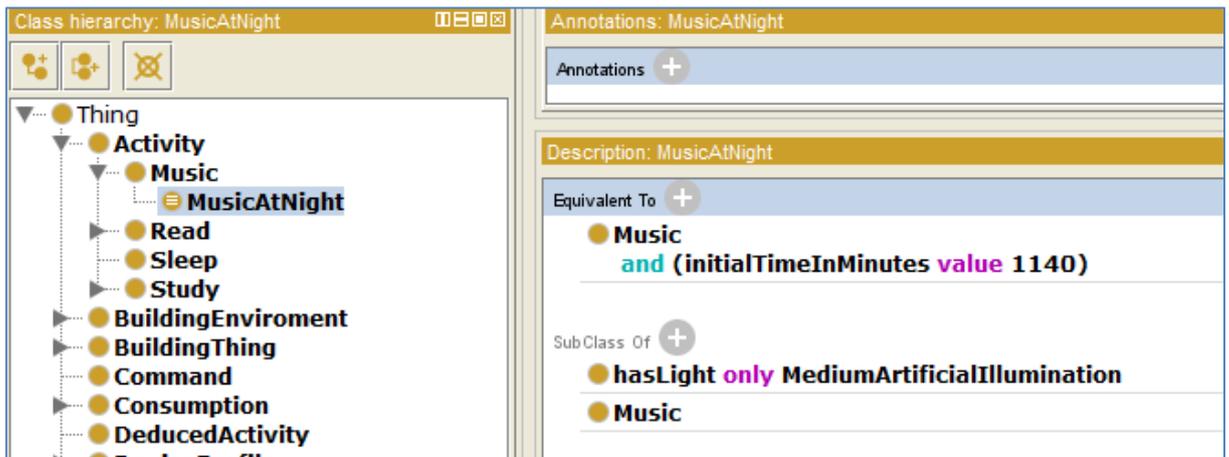


Ilustración 48: axioma MusicAtNight

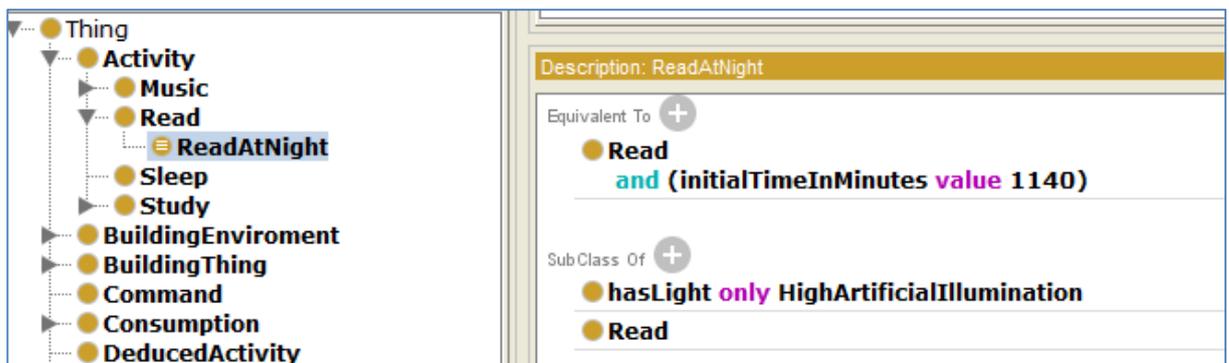


Ilustración 49: Axioma ReadAtNight

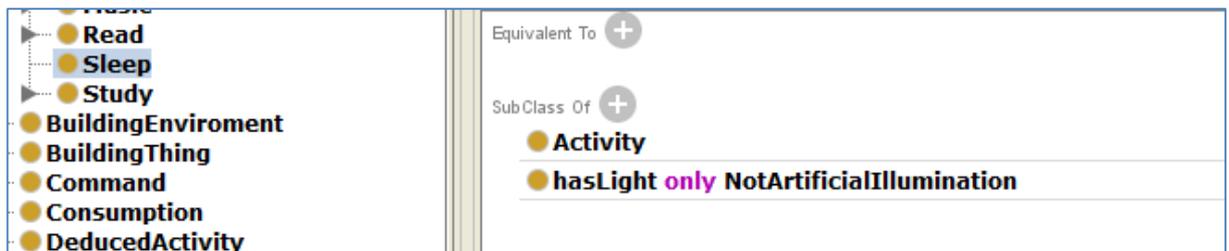


Ilustración 50: Axioma Sleep

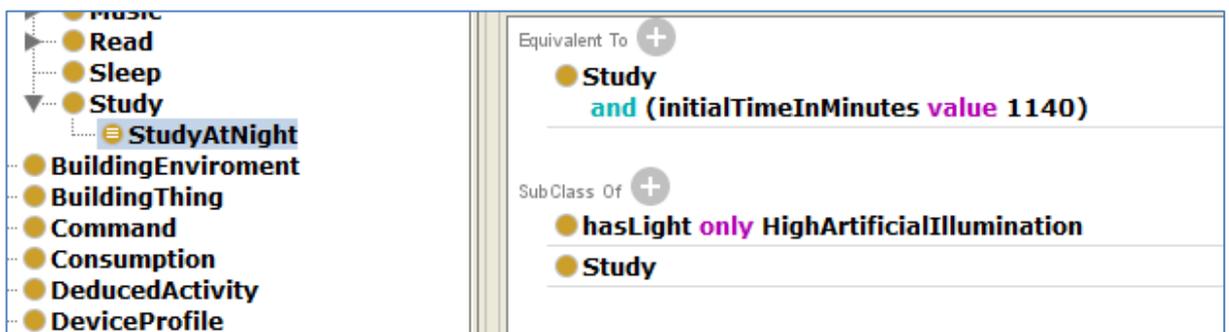


Ilustración 51: Axioma StudyAtNight

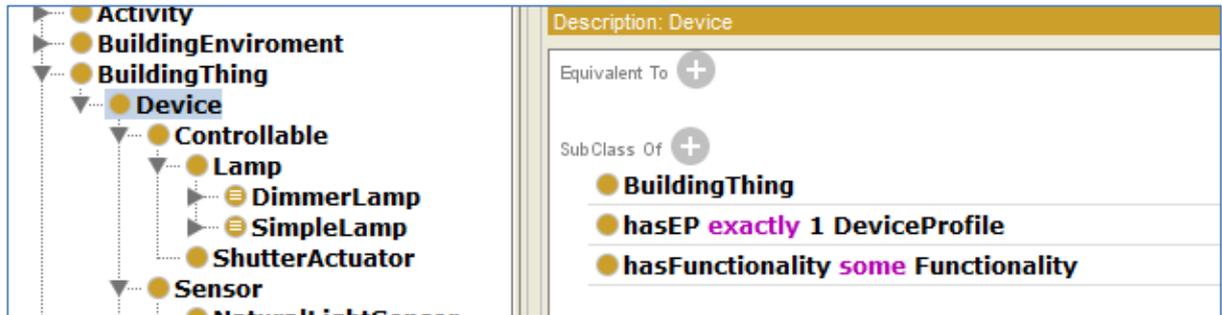


Ilustración 52: Axioma Device

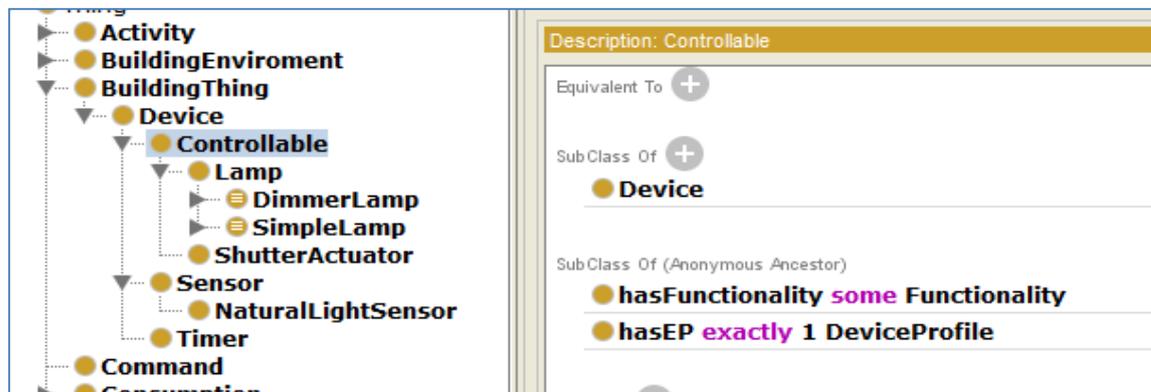


Ilustración 53: Axioma Controllable

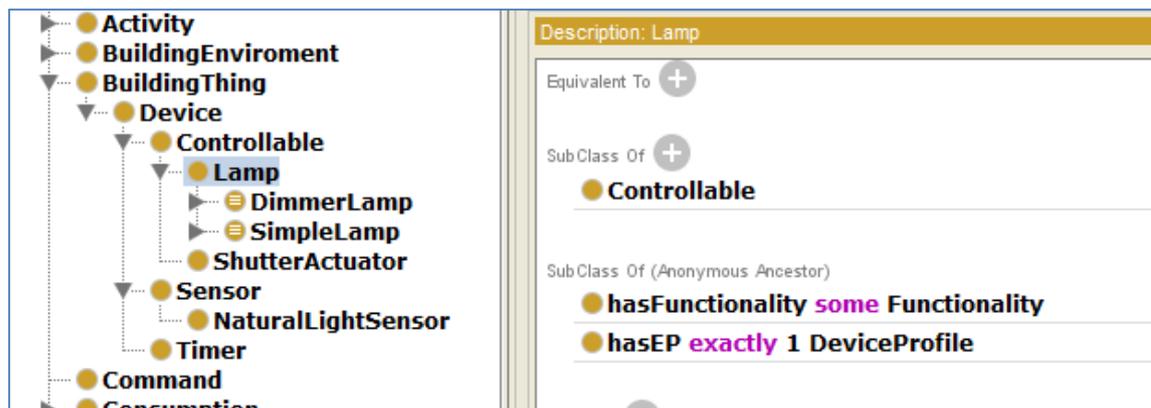


Ilustración 54: Axioma Lamp

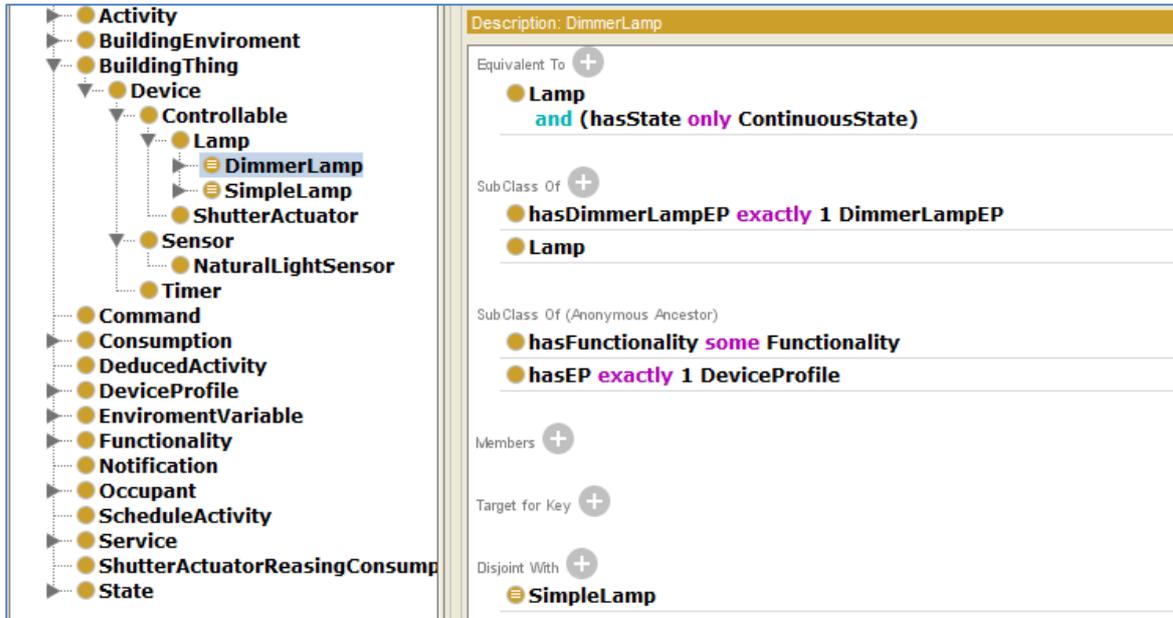


Ilustración 55: Axioma DimmerLamp

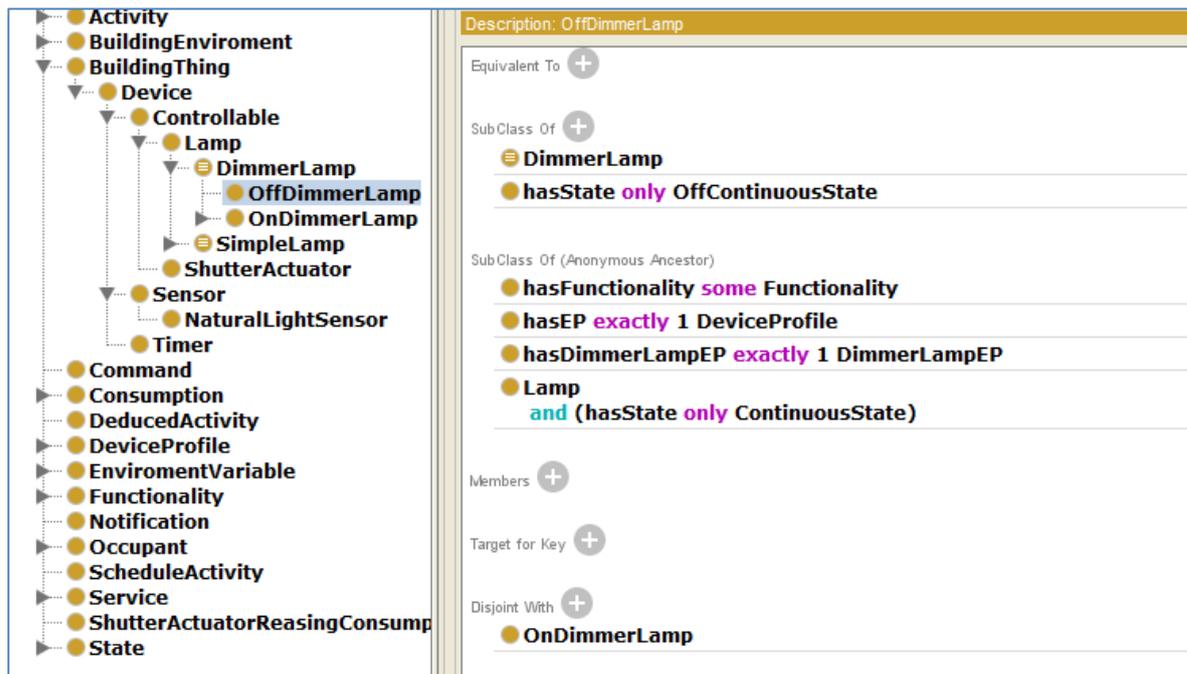


Ilustración 56: Axioma OffDimmerLamp

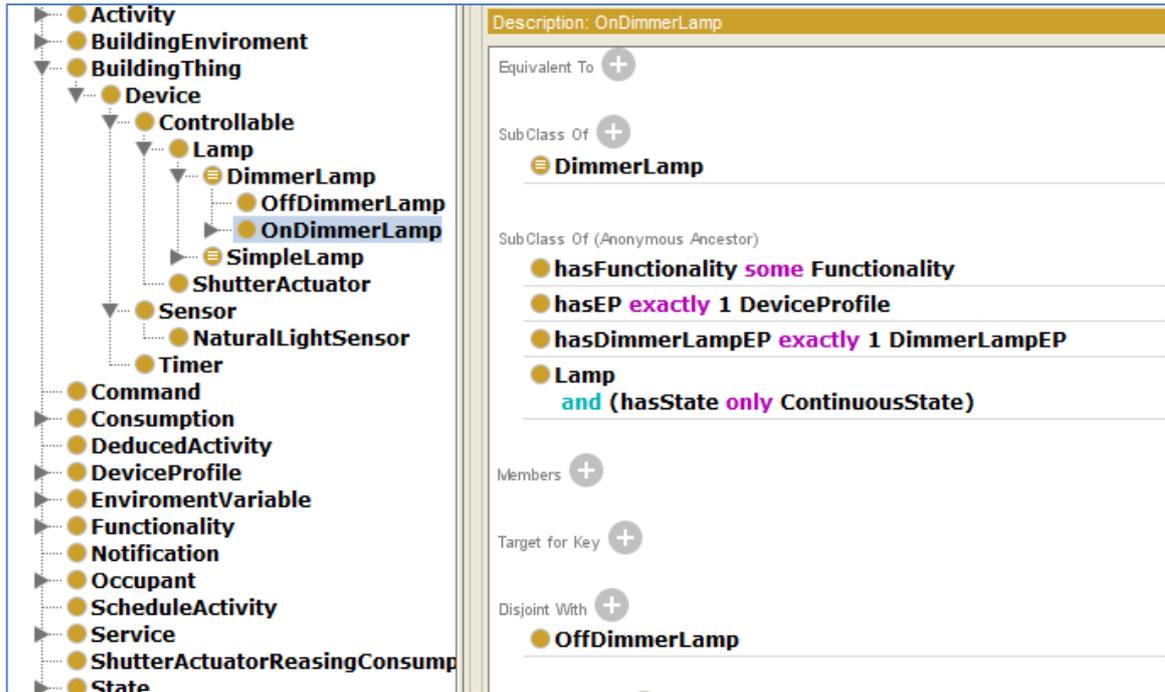


Ilustración 57: Axioma OnDimmerLamp

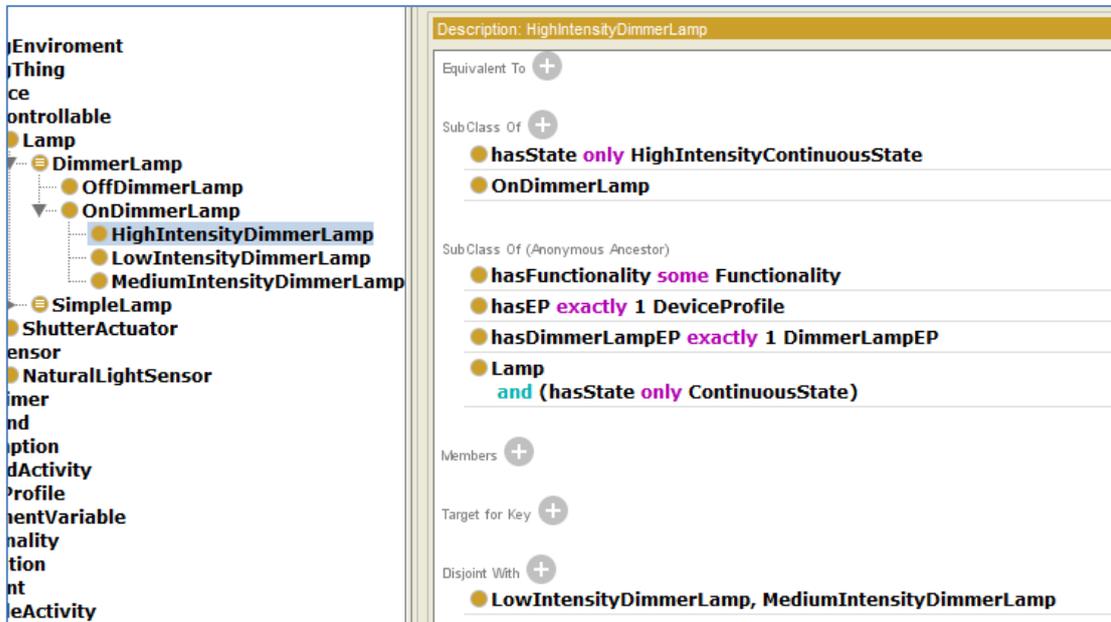


Ilustración 58: Axioma HighIntensityDimmerLamp

The screenshot displays an ontology editor interface. On the left, a class hierarchy is shown with 'LowIntensityDimmerLamp' selected. The right pane shows the description for 'LowIntensityDimmerLamp'.

**Description: LowIntensityDimmerLamp**

- Equivalent To: +
- SubClass Of: +
  - hasState **only** LowIntensityContinuousState
  - OnDimmerLamp
- SubClass Of (Anonymous Ancestor)
  - hasFunctionality **some** Functionality
  - hasEP **exactly 1** DeviceProfile
  - hasDimmerLampEP **exactly 1** DimmerLampEP
  - Lamp **and** (hasState **only** ContinuousState)
- Members: +
- Target for Key: +
- Disjoint With: +
  - HighIntensityDimmerLamp, MediumIntensityDimmerLamp

Ilustración 59: Axioma LowIntensityDimmerLamp

The screenshot displays an ontology editor interface. On the left, a class hierarchy is shown with 'MediumIntensityDimmerLamp' selected. The right pane shows the description for 'MediumIntensityDimmerLamp'.

**Description: MediumIntensityDimmerLamp**

- Equivalent To: +
- SubClass Of: +
  - hasState **only** MediumIntensityContinuousState
  - OnDimmerLamp
- SubClass Of (Anonymous Ancestor)
  - hasFunctionality **some** Functionality
  - hasEP **exactly 1** DeviceProfile
  - hasDimmerLampEP **exactly 1** DimmerLampEP
  - Lamp **and** (hasState **only** ContinuousState)
- Members: +
- Target for Key: +
- Disjoint With: +
  - LowIntensityDimmerLamp, HighIntensityDimmerLamp

Ilustración 60: Axioma MediumIntensityDimmerLamp

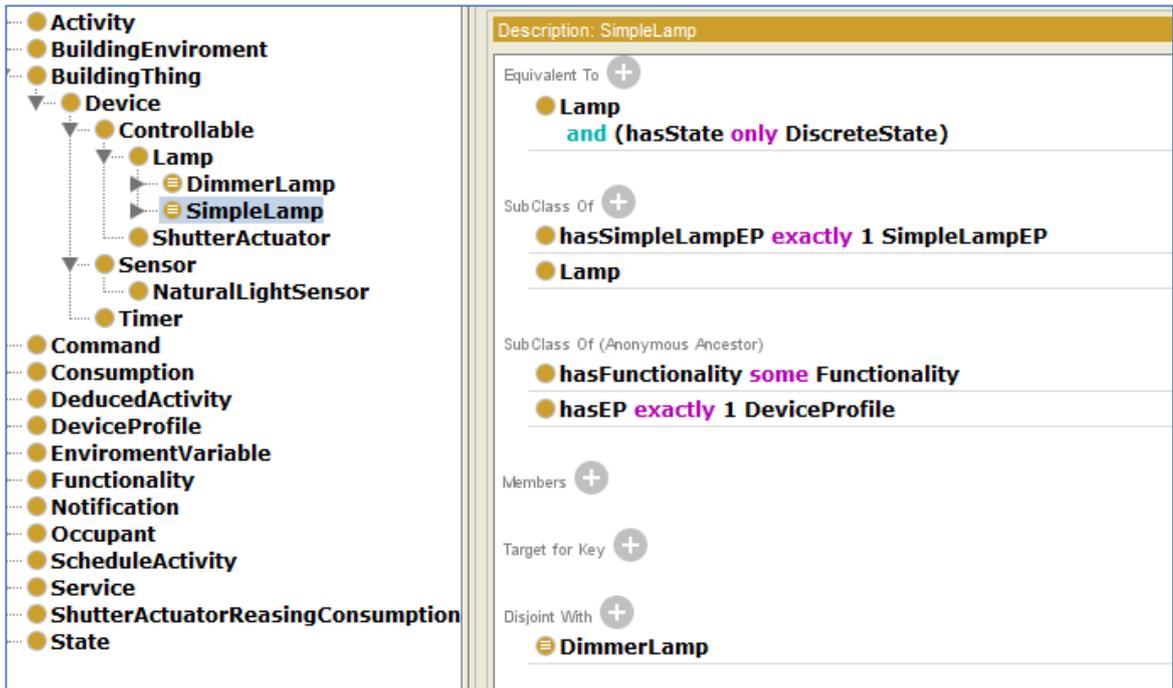


Ilustración 61: Axioma SimpleLamp

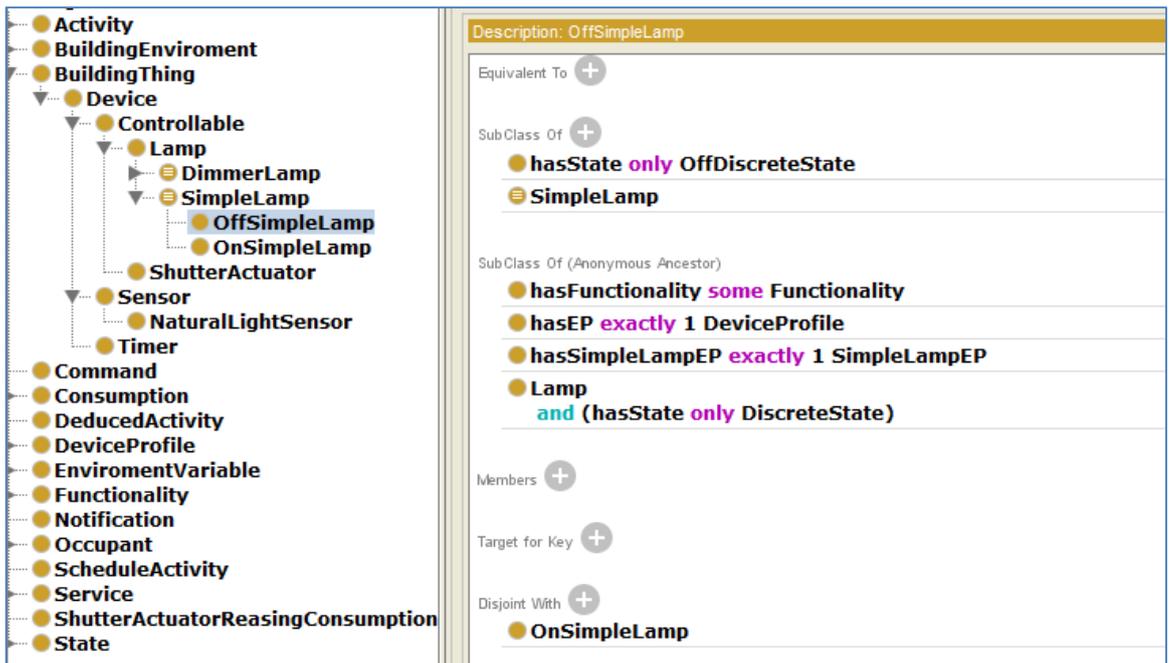


Ilustración 62: Axioma OffSimpleLamp

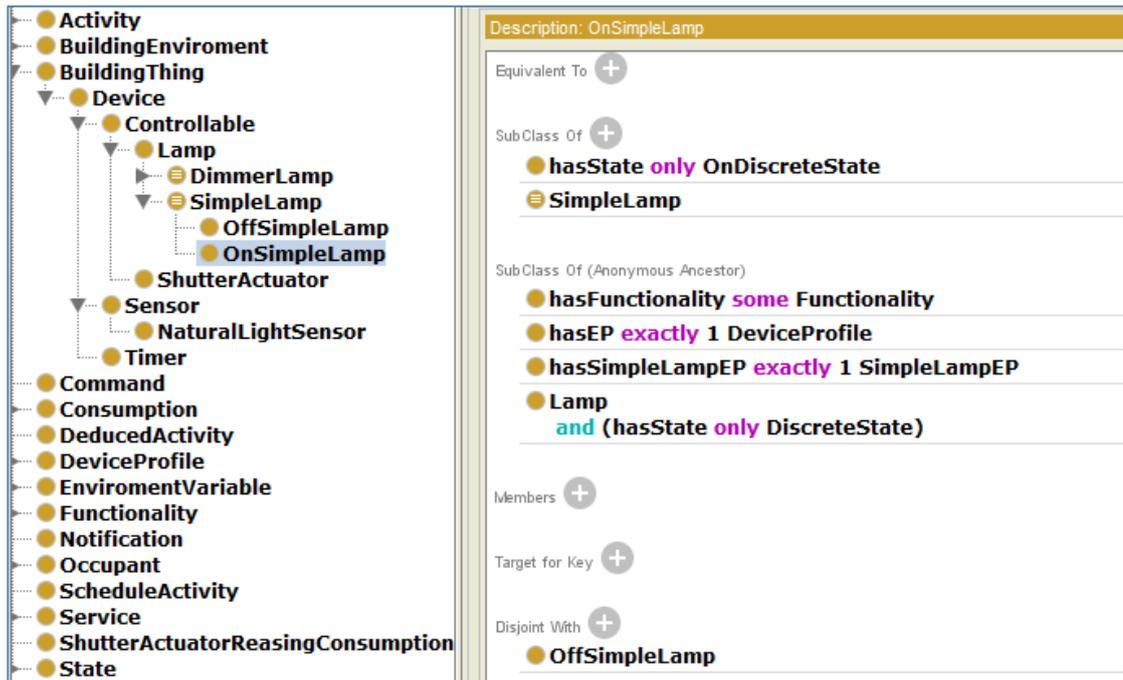


Ilustración 63: OnSimpleLamp

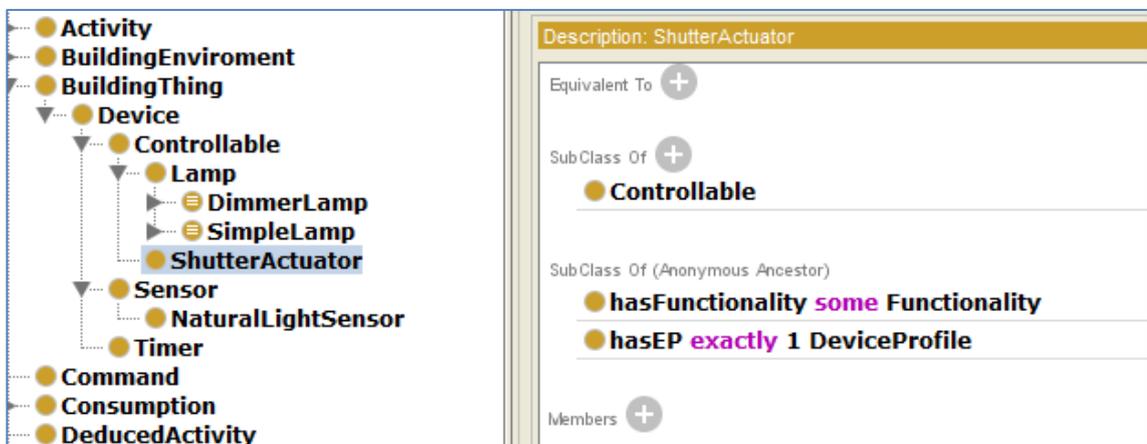


Ilustración 64: Axioma ShutterActuator

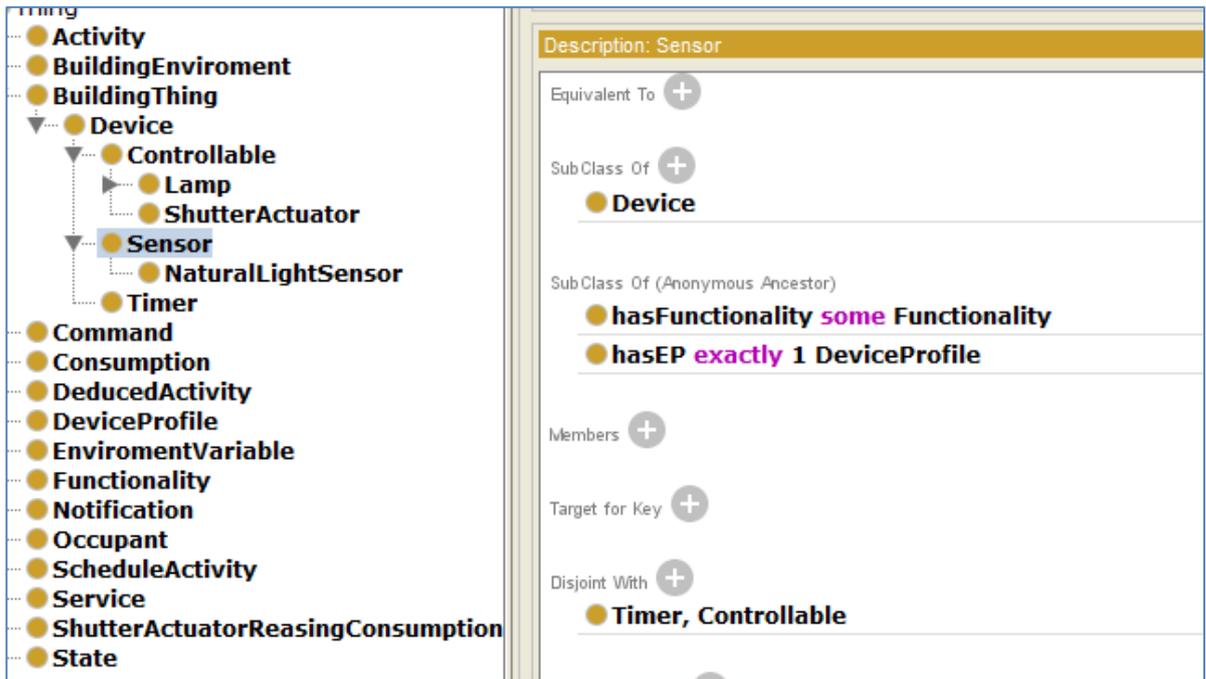


Ilustración 65: Axioma Sensor

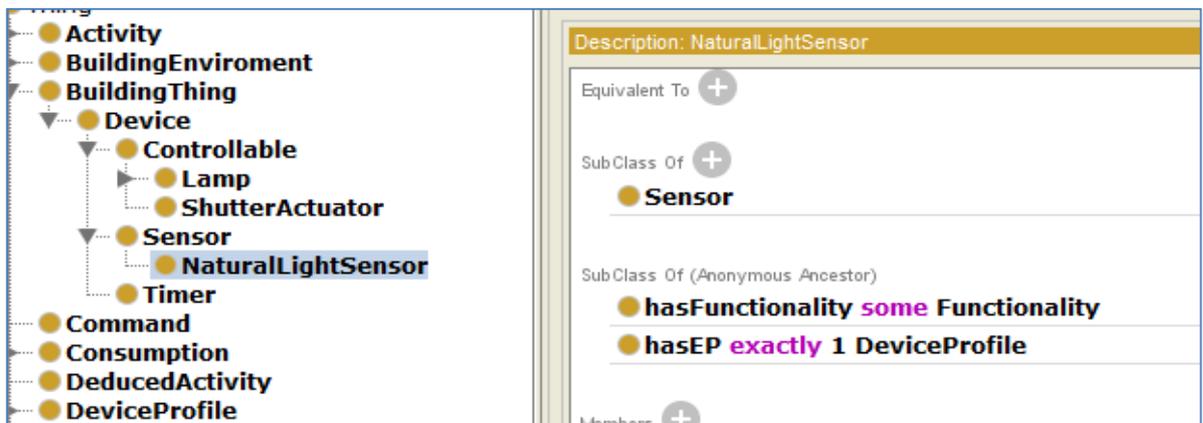


Ilustración 66: Axioma NaturalLightSensor

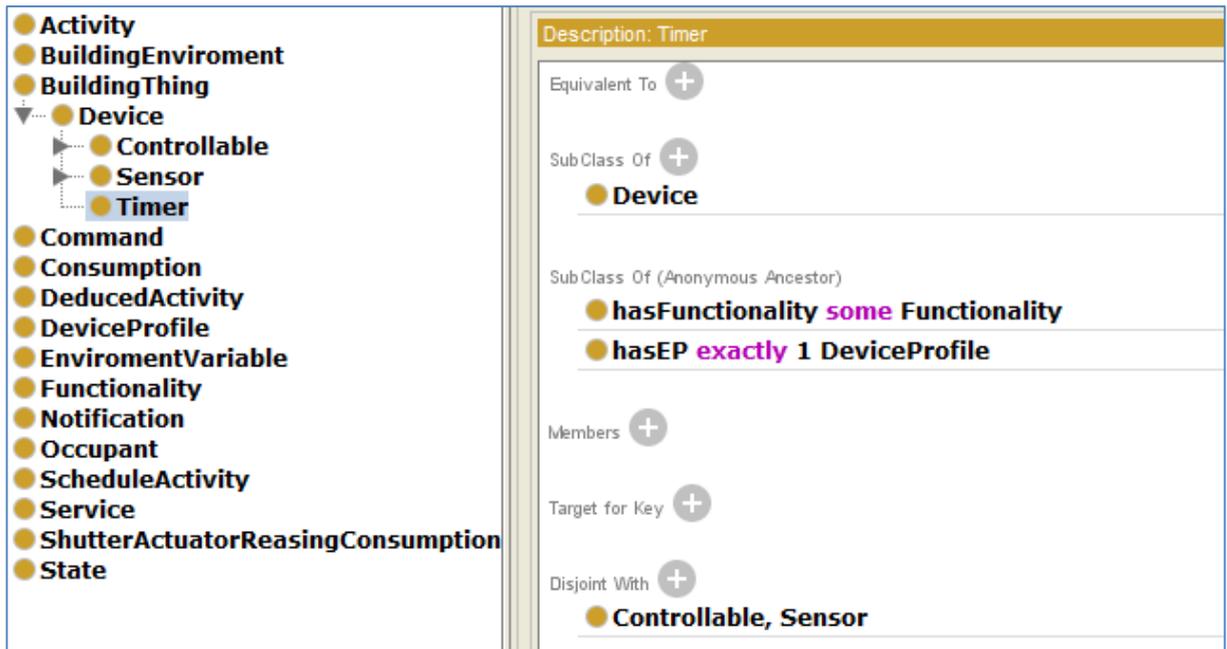


Ilustración 67: Axioma Timer

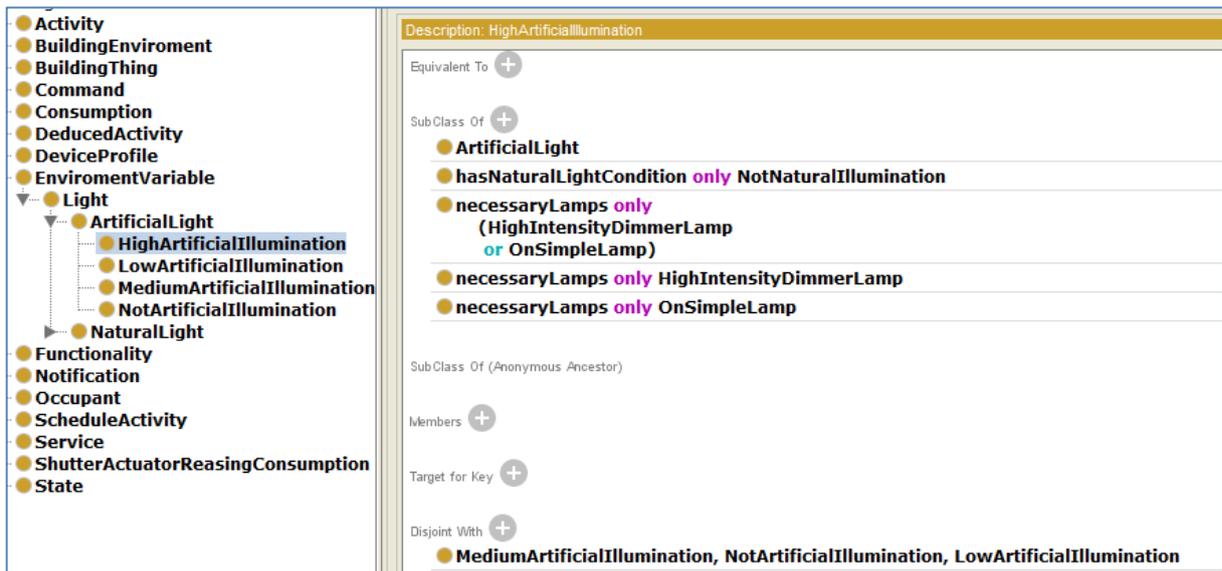


Ilustración 68: Axioma HighArtificialIllumination

**Class Hierarchy (Left Panel):**

- Activity
- BuildingEnvironment
- BuildingThing
- Command
- Consumption
- DeducedActivity
- DeviceProfile
- EnvironmentVariable
- Light
  - ArtificialLight
    - HighArtificialIllumination
    - LowArtificialIllumination
    - MediumArtificialIllumination
    - NotArtificialIllumination
  - NaturalLight
- Functionality
- Notification
- Occupant
- ScheduleActivity
- Service
- ShutterActuatorReasingConsumption
- State

**Class Details (Right Panel):**

**Description:** LowArtificialIllumination

Equivalent To: **ArtificialLight**

SubClass Of: **ArtificialLight**

SubClass Of (Anonymous Ancestor): **hasNaturalLightCondition only MediumNaturalIllumination**, **necessaryLamps only LowIntensityDimmerLamp**

Members: +

Target for Key: +

Disjoint With: **MediumArtificialIllumination, HighArtificialIllumination, NotArtificialIllumination**

Ilustración 69: Axioma LowArtificialIllumination

**Class Hierarchy (Left Panel):**

- Activity
- BuildingEnvironment
- BuildingThing
- Command
- Consumption
- DeducedActivity
- DeviceProfile
- EnvironmentVariable
- Light
  - ArtificialLight
    - HighArtificialIllumination
    - LowArtificialIllumination
    - MediumArtificialIllumination
    - NotArtificialIllumination
  - NaturalLight
- Functionality
- Notification
- Occupant
- ScheduleActivity
- Service
- ShutterActuatorReasingConsumption
- State

**Class Details (Right Panel):**

**Description:** MediumArtificialIllumination

Equivalent To: **ArtificialLight**

SubClass Of: **ArtificialLight**

SubClass Of (Anonymous Ancestor): **hasNaturalLightCondition only LowNaturalIllumination**, **necessaryLamps only MediumIntensityDimmerLamp**

Members: +

Target for Key: +

Disjoint With: **HighArtificialIllumination, NotArtificialIllumination, LowArtificialIllumination**

Ilustración 70: Axioma MediumArtificialIllumination

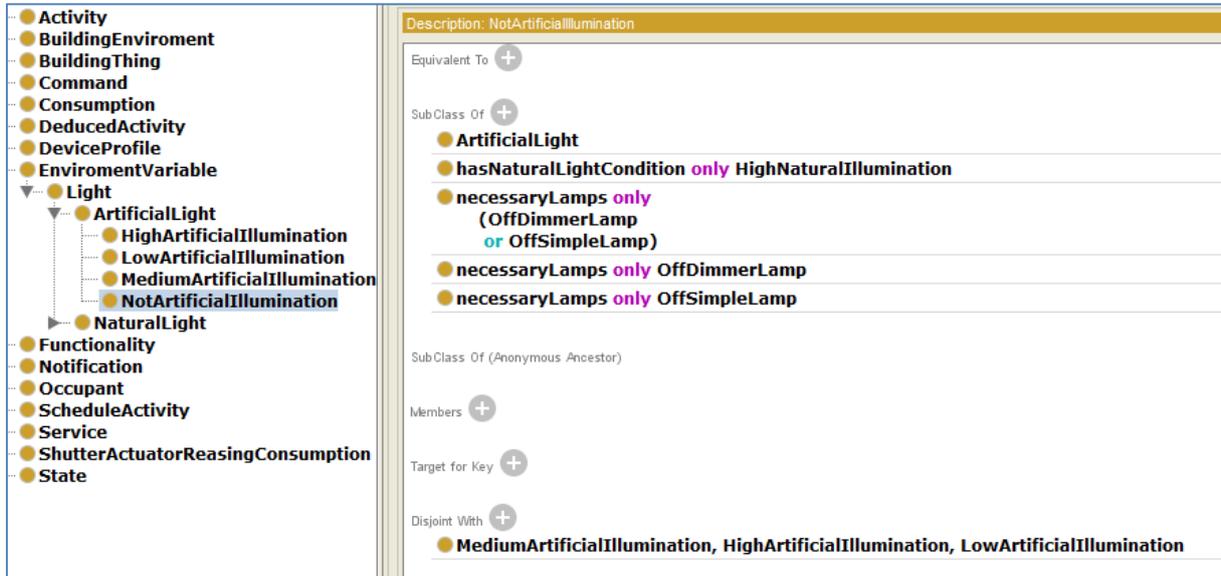


Ilustración 71: Axioma NotArtificialIllumination

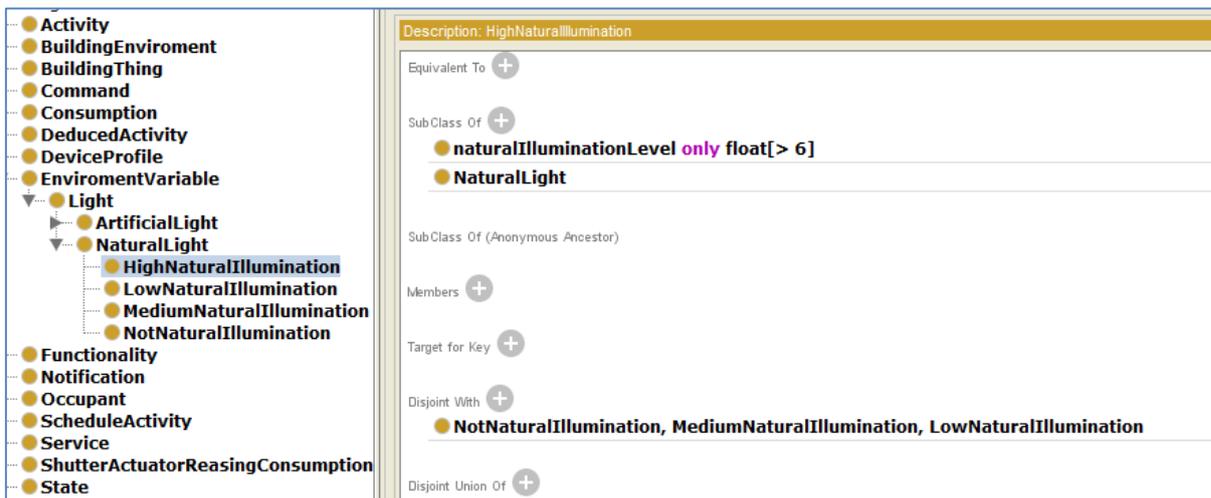


Ilustración 72: Axioma HighNaturalIllumination

● Activity  
 ● BuildingEnvironment  
 ● BuildingThing  
 ● Command  
 ● Consumption  
 ● DeducedActivity  
 ● DeviceProfile  
 ● EnvironmentVariable  
 ▼ ● Light  
 ● ArtificialLight  
 ▼ ● NaturalLight  
 ● HighNaturalIllumination  
 ● LowNaturalIllumination  
 ● MediumNaturalIllumination  
 ● NotNaturalIllumination  
 ● Functionality  
 ● Notification  
 ● Occupant  
 ● ScheduleActivity  
 ● Service  
 ● ShutterActuatorReasoningConsumption

Description: LowNaturalIllumination

Equivalent To +

Sub Class Of +  
 ● naturalIlluminationLevel only float[> 0, <= 3]  
 ● NaturalLight

Sub Class Of (Anonymous Ancestor)

Members +

Target for Key +

Disjoint With +  
 ● NotNaturalIllumination, MediumNaturalIllumination, HighNaturalIllumination

Ilustración 73: Axioma LowNaturalIllumination

● Activity  
 ● BuildingEnvironment  
 ● BuildingThing  
 ● Command  
 ● Consumption  
 ● DeducedActivity  
 ● DeviceProfile  
 ● EnvironmentVariable  
 ▼ ● Light  
 ● ArtificialLight  
 ▼ ● NaturalLight  
 ● HighNaturalIllumination  
 ● LowNaturalIllumination  
 ● MediumNaturalIllumination  
 ● NotNaturalIllumination  
 ● Functionality  
 ● Notification  
 ● Occupant  
 ● ScheduleActivity  
 ● Service  
 ● ShutterActuatorReasoningConsumption  
 ● State

Description: MediumNaturalIllumination

Equivalent To +

Sub Class Of +  
 ● naturalIlluminationLevel only float[> 3, <= 6]  
 ● NaturalLight

Sub Class Of (Anonymous Ancestor)

Members +

Target for Key +

Disjoint With +  
 ● NotNaturalIllumination, LowNaturalIllumination, HighNaturalIllumination

Disjoint Union Of +

Ilustración 74: Axioma MediumNaturalIllumination

**Activity**

- BuildingEnvironment
- BuildingThing
- Command
- Consumption
- DeducedActivity
- DeviceProfile
- EnvironmentVariable
- Light
  - ArtificialLight
  - NaturalLight
    - HighNaturalIllumination
    - LowNaturalIllumination
    - MediumNaturalIllumination
    - NotNaturalIllumination**
- Functionality
- Notification
- Occupant
- ScheduleActivity
- Service
- ShutterActuatorReasingConsumption
- State

**Description: NotNaturalIllumination**

Equivalent To +

Sub Class Of +

- naturalIlluminationLevel value 0.0f**
- NaturalLight**

Sub Class Of (Anonymous Ancestor)

Members +

Target for Key +

Disjoint With +

- MediumNaturalIllumination, LowNaturalIllumination, HighNaturalIllumination**

Disjoint Union Of +

Ilustración 75: Axioma NotNaturalIllumination

**Activity**

- BuildingEnvironment
- BuildingThing
- Command
- Consumption
- DeducedActivity
- DeviceProfile
- EnvironmentVariable
- Functionality
  - ControlFunctionality**
  - NotificationFunctionality
  - QueryFunctionality
- Notification
- Occupant
- ScheduleActivity
- Service
- ShutterActuatorReasingConsumption
- State

**Description: ControlFunctionality**

Equivalent To +

Sub Class Of +

- Functionality**
- hasCommand some Command**

Sub Class Of (Anonymous Ancestor)

Members +

Target for Key +

Disjoint With +

- QueryFunctionality, NotificationFunctionality**

Disjoint Union Of +

Ilustración 76: Axioma ControlFunctionality

The screenshot displays a class hierarchy on the left and a detailed description of the **NotificationFunctionality** class on the right.

**Class Hierarchy (Left):**

- Activity
- BuildingEnvironment
- BuildingThing
- Command
- Consumption
- DeducedActivity
- DeviceProfile
- EnvironmentVariable
- Functionality
  - ControlFunctionality
  - NotificationFunctionality**
  - QueryFunctionality
- Notification
- Occupant
- ScheduleActivity
- Service
- ShutterActuatorReasingConsump
- State

**NotificationFunctionality Description (Right):**

- Equivalent To: +
- SubClass Of: +
  - Functionality
  - hasNotification some Notification
- SubClass Of (Anonymous Ancestor):
- Members: +
- Target for Key: +
- Disjoint With: +
  - QueryFunctionality, ControlFunctionality
- Disjoint Union Of: +

Ilustración 77: Axioma NotificationFunctionality

The screenshot displays a class hierarchy on the left and a detailed description of the **NocturnMusicListener** class on the right.

**Class Hierarchy (Left):**

- Thing
  - Activity
  - BuildingEnvironment
  - BuildingThing
  - Command
  - Consumption
  - DeducedActivity
  - DeviceProfile
  - EnvironmentVariable
  - Functionality
  - Notification
  - Occupant
    - NocturnMusicListener
    - NocturnReader
    - NocturnStudent
  - ScheduleActivity
  - Service

**NocturnMusicListener Description (Right):**

- Equivalent To: +
  - Occupant
  - and (hasPreference only MusicAtNight)
- SubClass Of: +
  - Occupant
- SubClass Of (Anonymous Ancestor):
- Members: +
- Target for Key: +

Ilustración 78: Axioma NocturnMusicListener

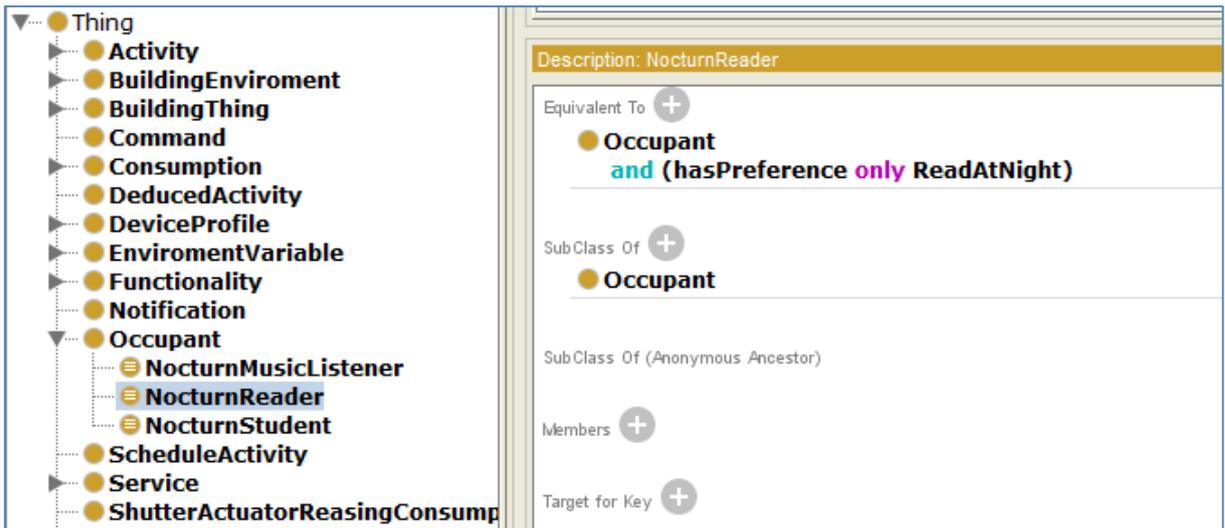


Ilustración 79: Axioma NocturnReader

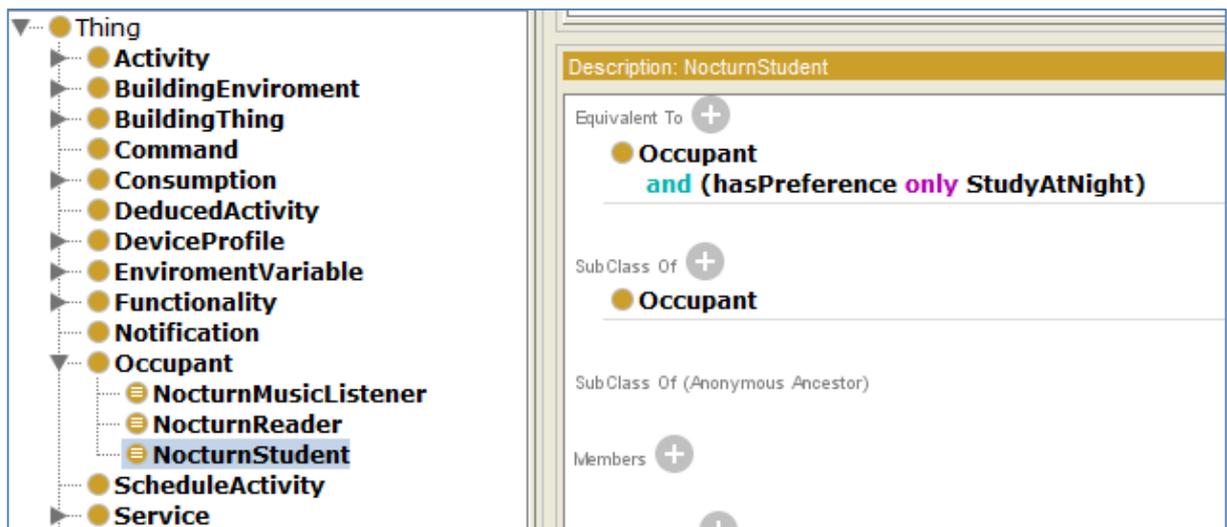


Ilustración 80: Axioma NocturnStudent

**Activity**  
**BuildingEnvironment**  
**BuildingThing**  
**Command**  
**Consumption**  
**DeducedActivity**  
**DeviceProfile**  
**EnvironmentVariable**  
**Functionality**  
**Notification**  
**Occupant**  
**ScheduleActivity**  
**Service**  
**ShutterActuatorReasingConsumption**  
**State**  
 ● **ContinuousState**  
   ● **HighIntensityContinuousState**  
   ● **LowIntensityContinuousState**  
   ● **MediumIntensityContinuousState**  
   ● **OffContinuousState**  
 ⊖ **DiscreteState** ≡ **ShutterState**  
 ⊖ **ShutterState** ≡ **DiscreteState**

**Description: HighIntensityContinuousState**

Equivalent To +

SubClass Of +  
 ● **ContinuousState**  
 ● **hasContinuousStateValue only float[> 6 , <= 10]**

SubClass Of (Anonymous Ancestor)

Members +

Target for Key +

Disjoint With +  
 ● **LowIntensityContinuousState, OffContinuousState, MediumIntensityContinuousState**

Disjoint Union Of +

Ilustración 81: Axioma HighIntensityContinuousState

**Activity**  
**BuildingEnvironment**  
**BuildingThing**  
**Command**  
**Consumption**  
**DeducedActivity**  
**DeviceProfile**  
**EnvironmentVariable**  
**Functionality**  
**Notification**  
**Occupant**  
**ScheduleActivity**  
**Service**  
**ShutterActuatorReasingConsumption**  
**State**  
 ● **ContinuousState**  
   ● **HighIntensityContinuousState**  
   ● **LowIntensityContinuousState**  
   ● **MediumIntensityContinuousState**  
   ● **OffContinuousState**  
 ⊖ **DiscreteState** ≡ **ShutterState**  
 ⊖ **ShutterState** ≡ **DiscreteState**

**Description: LowIntensityContinuousState**

Equivalent To +

SubClass Of +  
 ● **ContinuousState**  
 ● **hasContinuousStateValue only float[> 0 , <= 3]**

SubClass Of (Anonymous Ancestor)

Members +

Target for Key +

Disjoint With +  
 ● **OffContinuousState, HighIntensityContinuousState, MediumIntensityContinuousState**

Disjoint Union Of +

Ilustración 82: Axioma LowIntensityContinuousState

**Description: MediumIntensityContinuousState**

Equivalent To +

SubClass Of +

- ContinuousState
- hasContinuousSateValue **only** float[> 3 , <= 6]

SubClass Of (Anonymous Ancestor)

Members +

Target for Key +

Disjoint With +

- LowIntensityContinuousState, OffContinuousState, HighIntensityContinuousState

Disjoint Union Of +

Ilustración 83: Axioma MediumIntensityContinuousState

**Description: OffContinuousState**

Equivalent To +

SubClass Of +

- ContinuousState
- hasContinuousSateValue **value** 0.0f

SubClass Of (Anonymous Ancestor)

Members +

Target for Key +

Disjoint With +

- LowIntensityContinuousState, HighIntensityContinuousState, MediumIntensityContinuousState

Disjoint Union Of +

Ilustración 84: Axioma OffContinuousState

The screenshot displays a software interface with two main panes. The left pane shows a class hierarchy tree with the following items: Notification, Occupant, ScheduleActivity, Service, ShutterActuatorReasingConsumption, State, ContinuousState, DiscreteState ≡ ShutterState, and ShutterState ≡ DiscreteState. The right pane, titled 'Description: DiscreteState', contains the following information:

- Equivalent To: ShutterState
- SubClass Of: State
- SubClass Of (Anonymous Ancestor): DiscreteState
- Members: (empty)
- Target for Key: (empty)
- Disjoint With: ContinuousState

Ilustración 85: Axioma DiscreteState

The screenshot displays a software interface with two main panes. The left pane shows a class hierarchy tree with the following items: EnviromentVariable, Functionality, Notification, Occupant, ScheduleActivity, Service, ShutterActuatorReasingConsumption, State, ContinuousState, DiscreteState ≡ ShutterState, OffDiscreteState, OnDiscreteState, and ShutterState ≡ DiscreteState. The right pane, titled 'Description: OffDiscreteState', contains the following information:

- Equivalent To: (empty)
- SubClass Of: DiscreteState
- SubClass Of (Anonymous Ancestor): ShutterState, DiscreteState
- Members: hasDiscreteStateValue value false
- Target for Key: (empty)
- Disjoint With: OnDiscreteState

Ilustración 86: Axioma OffDiscreteState

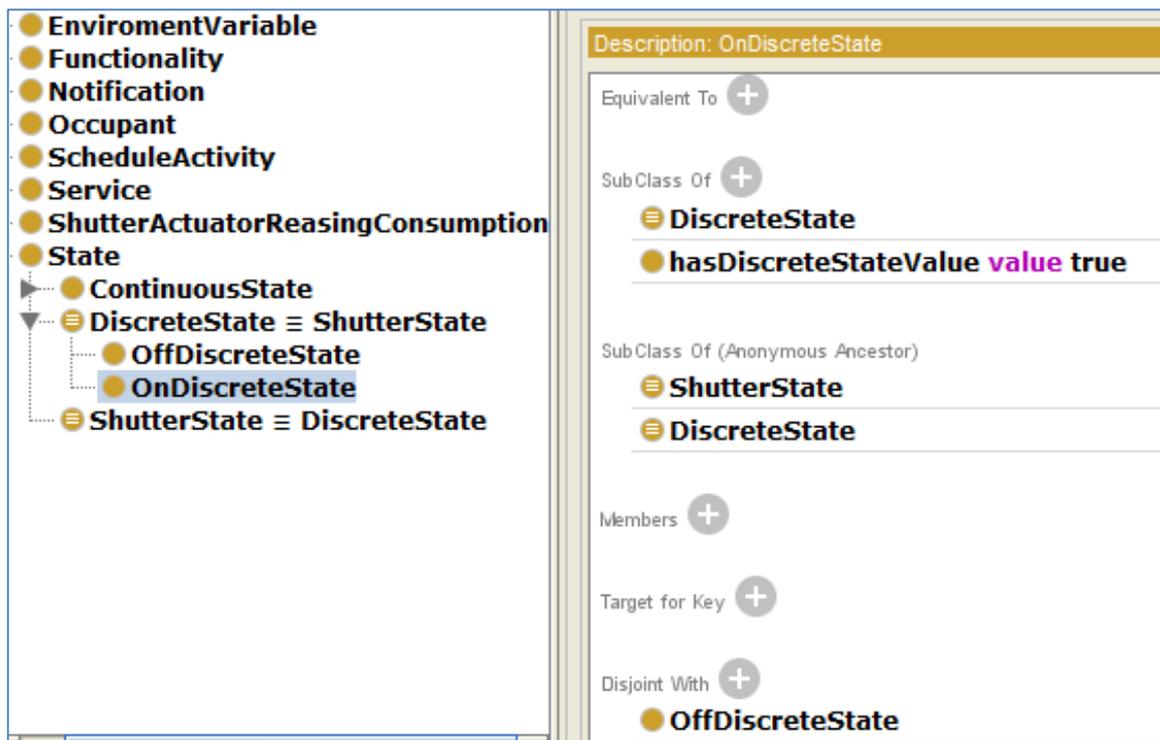


Ilustración 87: Axioma OnDiscreteState

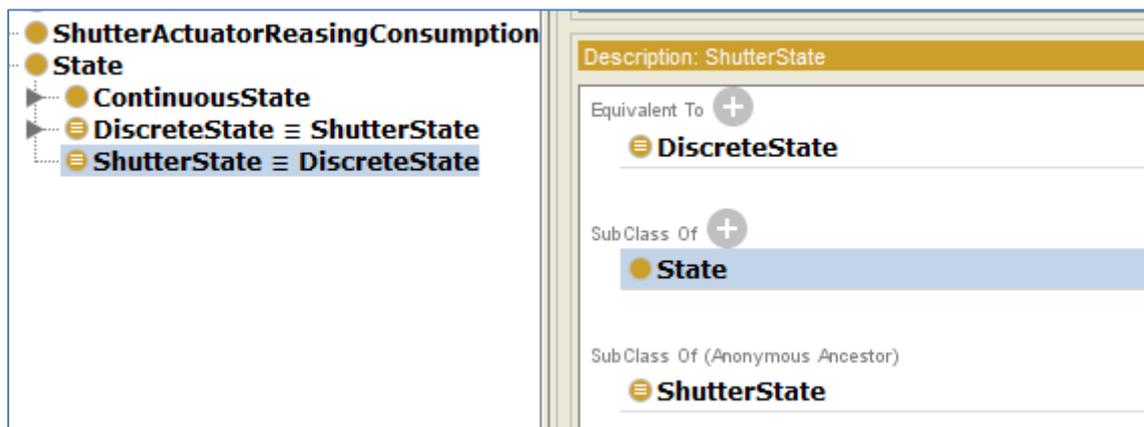


Ilustración 88: Axioma ShutterState

## CREACIÓN DE LAS REGLAS DE INFERENCIA DE LA ONTOLOGÍA

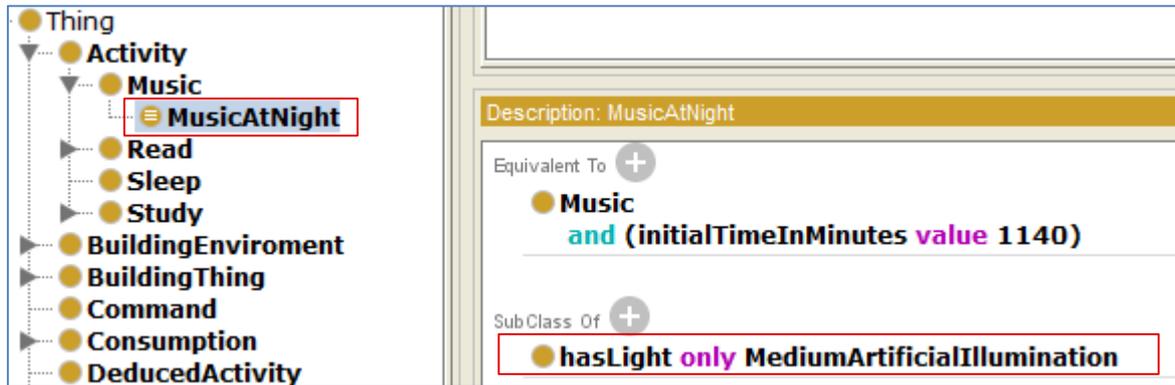


Ilustración 89: Regla de Inferencia MusicAtNight



Ilustración 90: Regla de Inferencia ReadAtNight

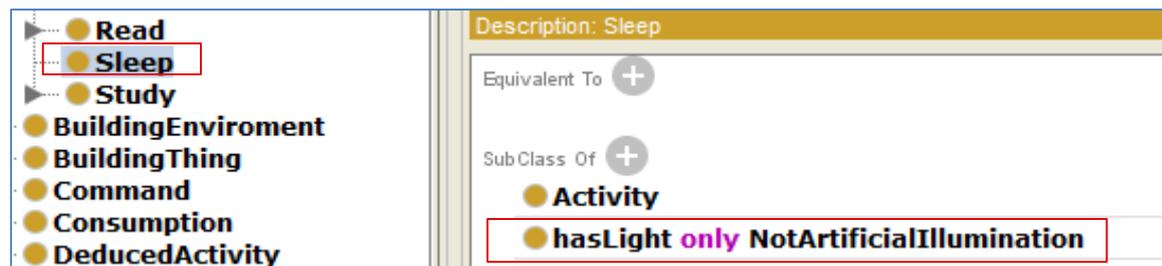


Ilustración 91: Regla de Inferencia Sleep

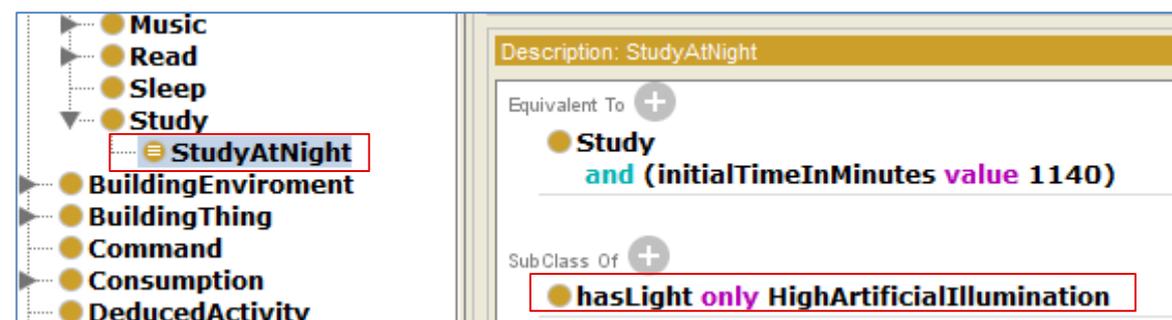


Ilustración 92: Regla de Inferencia StudyAtNight

Definir los datos de prueba a utilizar, los cuales se describen en la Tabla 7.

Tabla 7: Datos de prueba para el Ocupante

	<b>Occupant</b>
Age	30
ArrivalTime	'21:00:00-05:00'
LeaveTime	'05:00:00-05:00'
TimeOfSleep	'05:00:00-05:00'
TimeOfWakeUp	'05:00:00-05:00'

Luego se crean las instancias en Protégé: de tipo Occupant para Ocupante, y sus datos, tal como se observa en la ilustración 93.

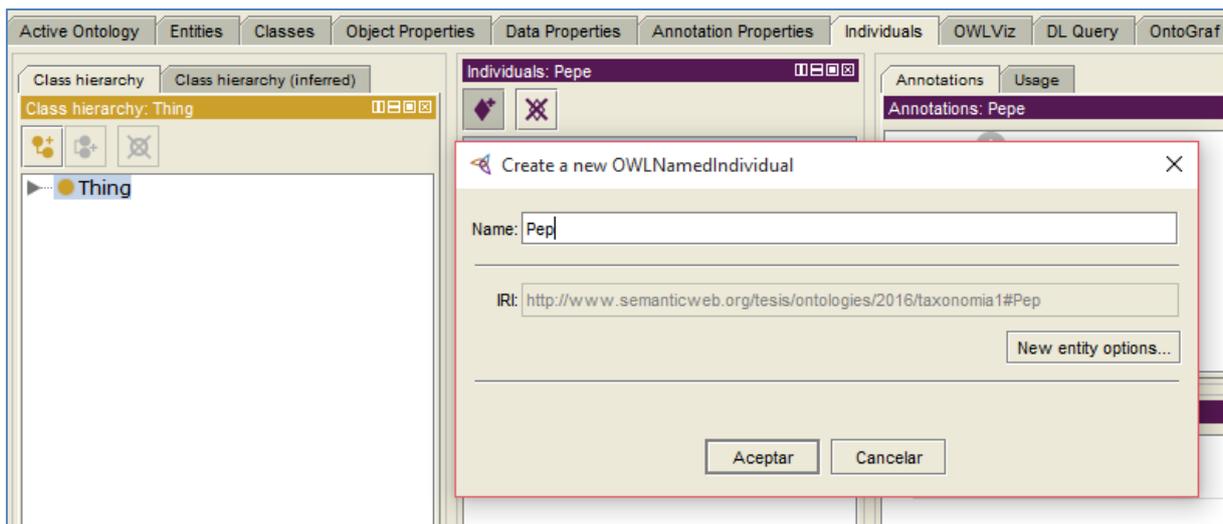


Ilustración 93: Creación Individuals - Occupant

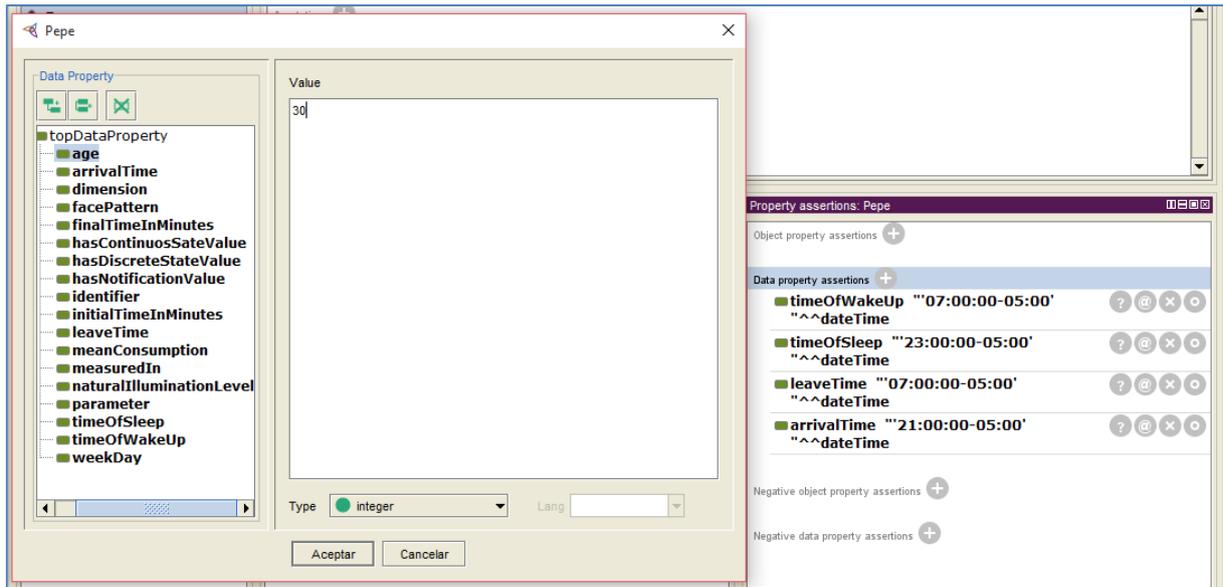


Ilustración 94: Creación property age para Pepe

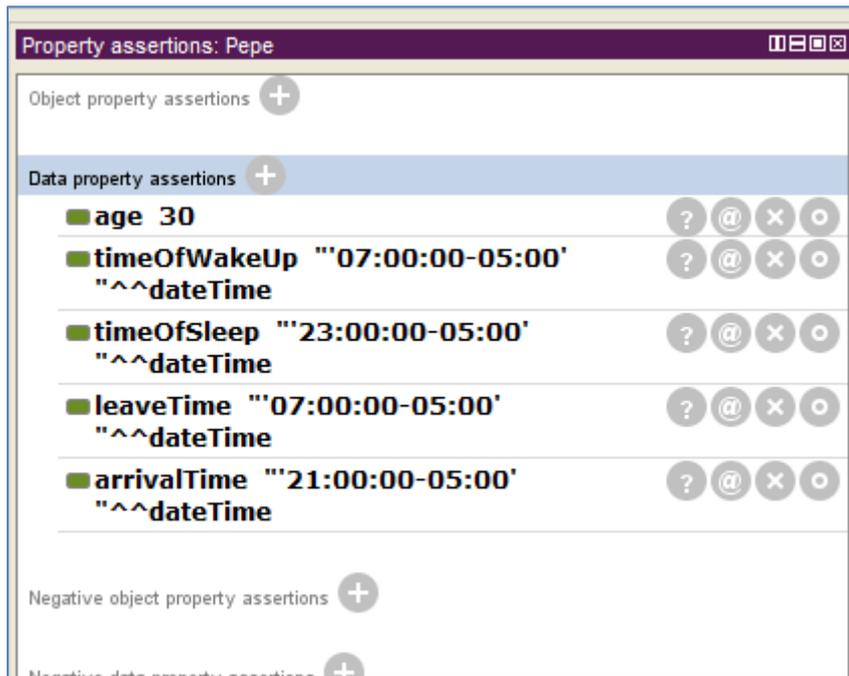


Ilustración 95: Property de Occupant Pepe

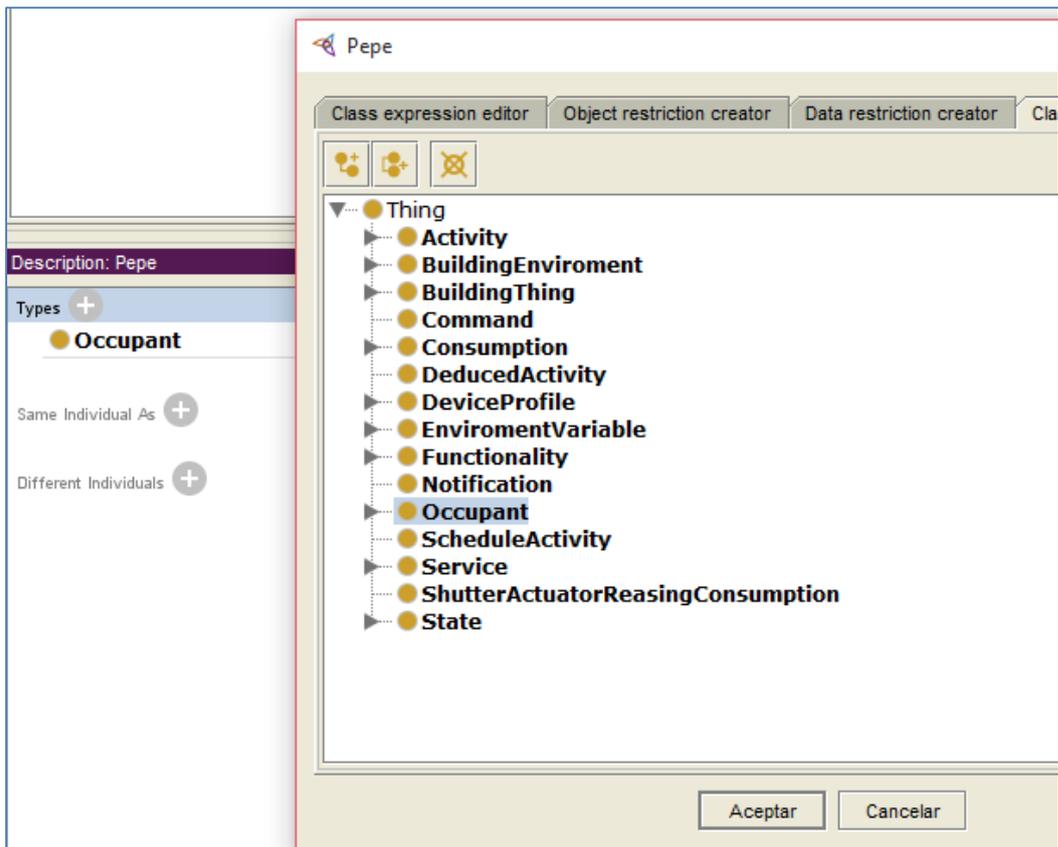


Ilustración 96: Escoger clase perteneciente a Pepe

## EJECUCIÓN EN EL RAZONADOR JENA 2.1 EN UN ENTORNO WEB

Luego de conectarse el servidor de Jena se procede a ingresar a través del navegador con el localhost: 3030. Con el procedimiento anterior, nos encontramos en la página principal del razonador donde se escogerá la opción de Control Panel, para el uso con la ontología.

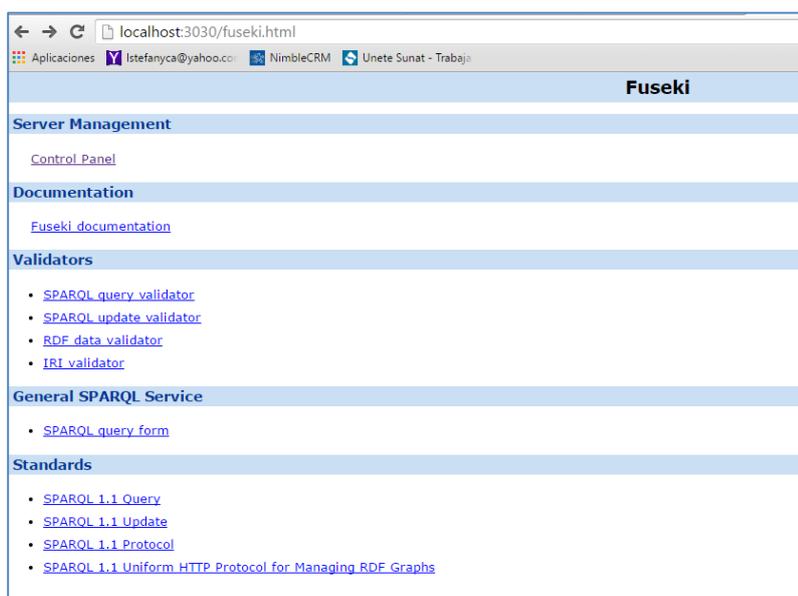


Ilustración 97: Menú Principal Razonador Jena

Para empezar a generar las consultas en el razonador es necesario ingresar la ontología desarrollada. En la parte inferior del menú de la página existe la opción de elegir la ontología como se muestra en la Ilustración 98.

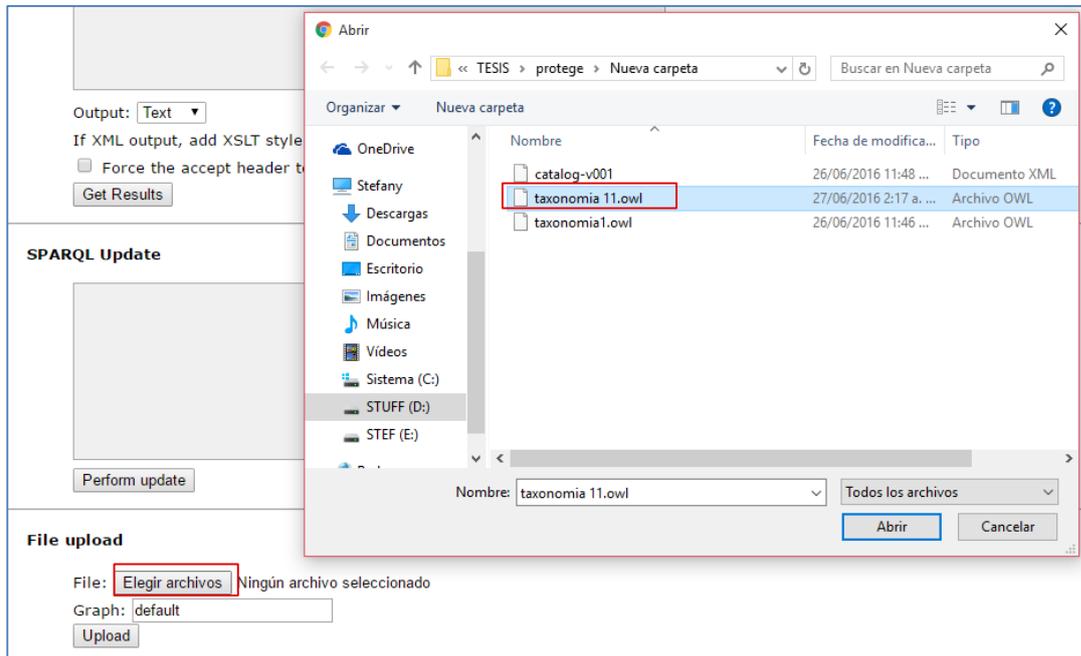


Ilustración 98: Ingreso de la Ontología

Mientras la ontología este correctamente estructurada, el razonador procede a ingresar los datos y genera el contador de Triples que la Ontología contiene. Como se muestra en la Ilustración 99 el resultado exitoso del ingreso de la Ontología.

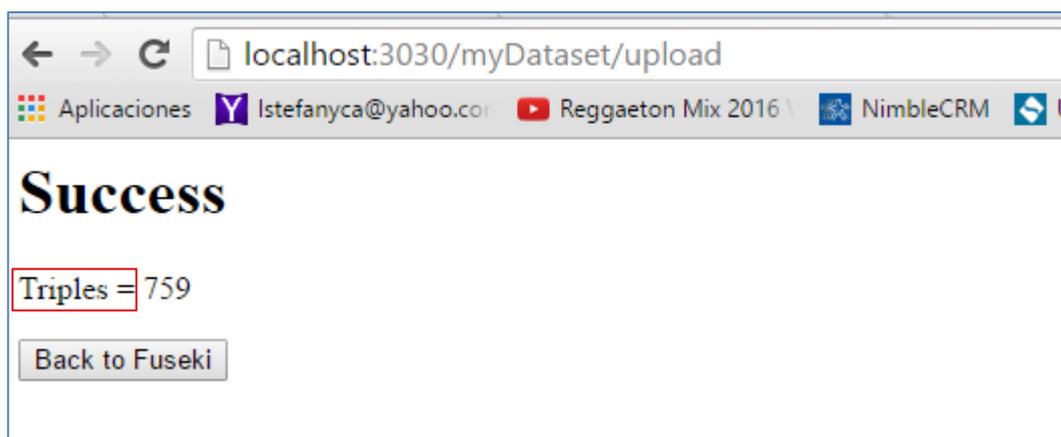


Ilustración 99: Ingreso exitoso ontología a razonador

Para comprobar que los triples sean correctos, se realiza una consulta general, donde seleccionamos todos los datos de la ontología en sujeto, predicado y objeto. Es decir los triples.

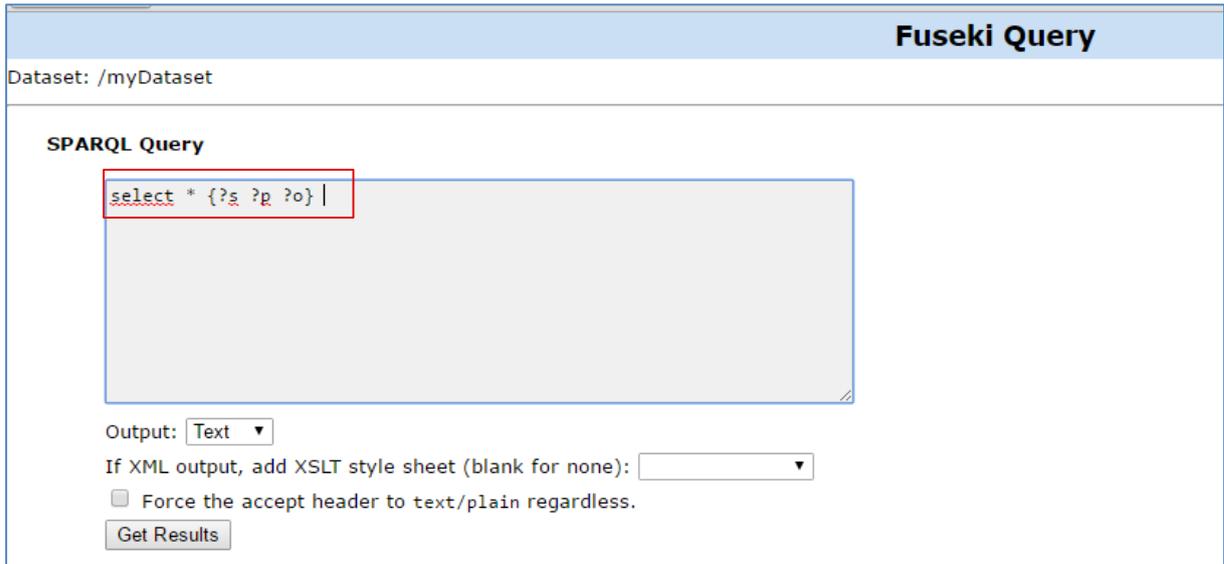


Ilustración 100: Sentencia general

Los resultados generados por la consulta se muestran en la siguiente Ilustración 101, indicando el sujeto, predicado y objeto.



Ilustración 101: Resultado Sentencia general

Como prueba se realiza una consulta básica para corroborar el uso debido de la ontología y el razonador.

- Music

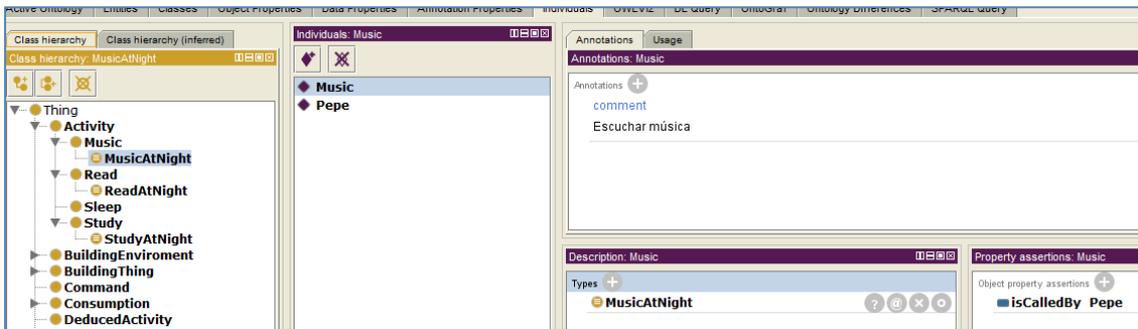


Ilustración 102: Music

- Read

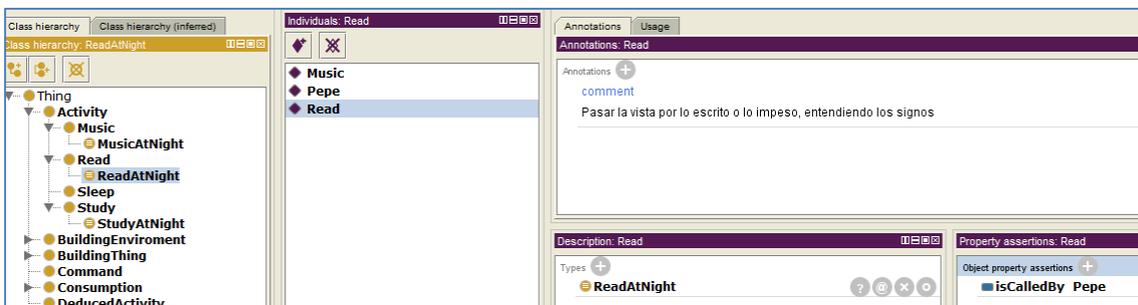


Ilustración 103: Read

- Study

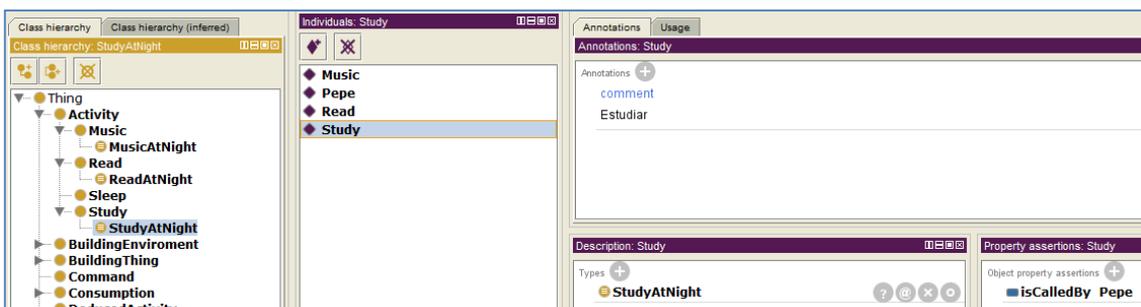


Ilustración 104: Study

## Creación de una consulta o Query para llamar a todas las acciones IsCalledBy Pepe



Ilustración 105: Query llamar todas acciones

En la Ilustración 106 podemos observar el resultado dado por la query anteriormente generada, que son todas las acciones que el sujeto Pepe puede ejecutar.

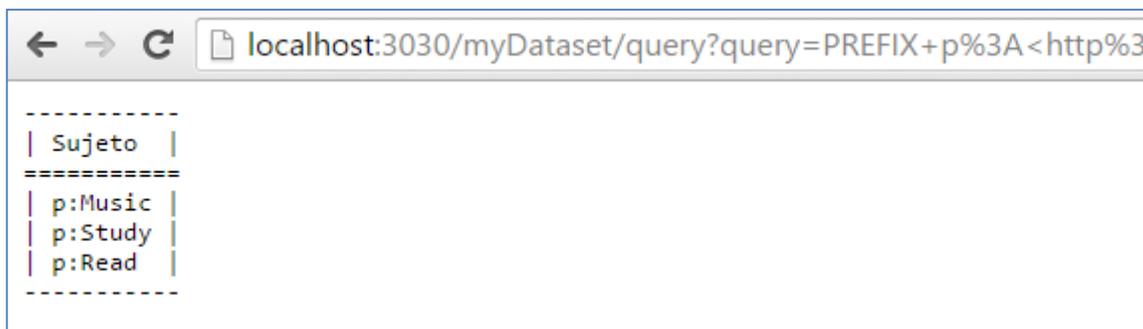


Ilustración 106: Resultado Query todas acciones

## Creación de una consulta o Query para llamar al sujeto padre de MusicAtNight

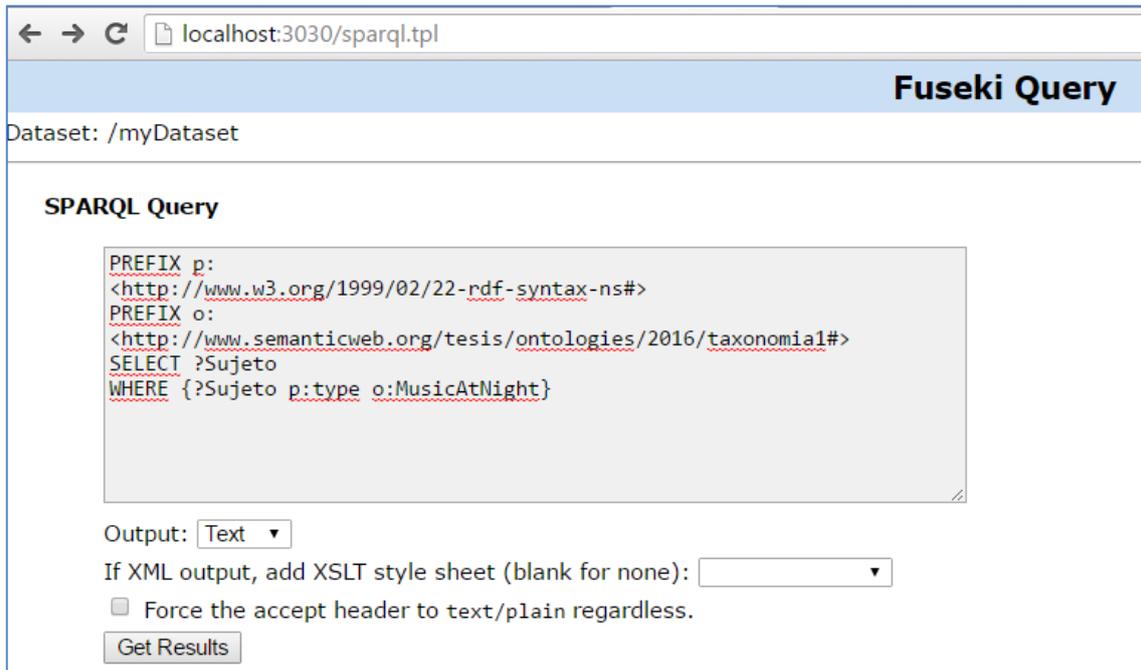


Ilustración 107: Query sujeto padre MusicAtNight

En la Ilustración 108, podemos observar el resultado dado por la query anteriormente generada, que mostrará al padre de la acción MusicAtNight.

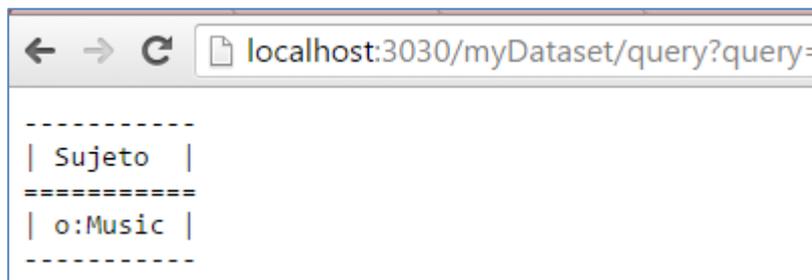


Ilustración 108: Resultado Query MusicAtNight

## CAPITULO V: DISCUSION DE RESULTADOS

El presente capítulo se realiza el análisis de los resultados a través de creación de individuals en el software Protegé 5.0 y el rasonador Pellet.

Para corroborar este capítulo nos basaremos en los resultados obtenidos por cada Regla de Inferencia anteriormente propuestas, enfocadas en las actividades nocturnas.

### REGLA DE INFERENCIA N° 1

- **Si Occupant is MusicAtNight Entonces Light MediumArtificialIllumination.**

Se crea el primer individual Occupant1 haciendo referencia a su tipo “NocturnMusicListener”, los atributos name y age, además de agregar su Object property en este caso “hasPreference” hacia Music1, con el propósito de relacionar al Occupant1 con la preferencia de Music1.

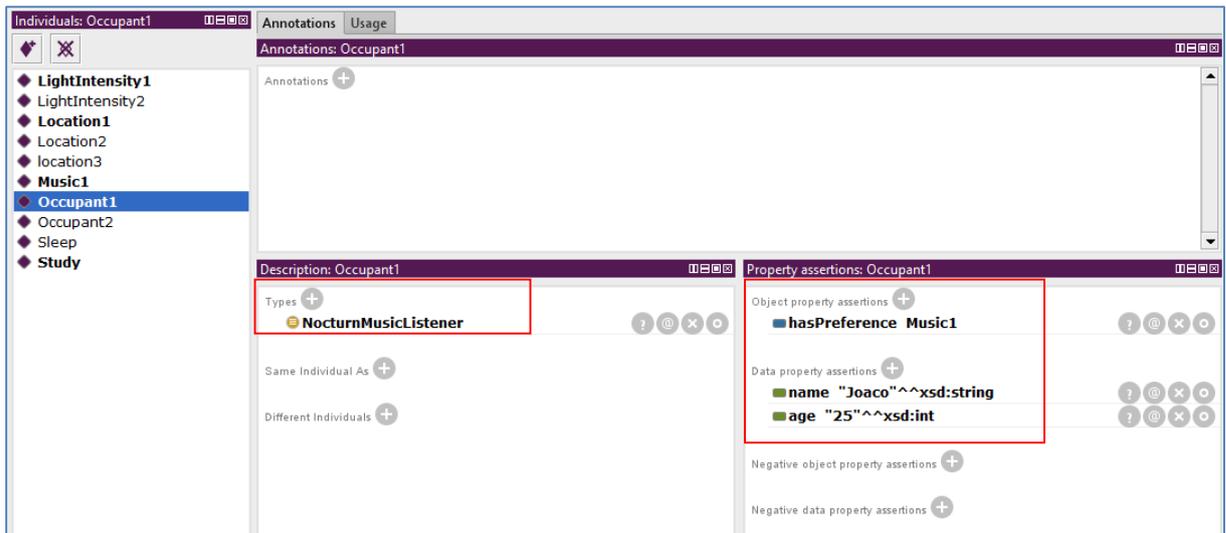


Ilustración 109: Creación Occupant1

Se crea el segundo individual Location1 haciendo referencia a su tipo “Location”, los atributos name, además de agregar su Object property en este caso “hasOccupant” hacia Occupant1, con el propósito de relacionar la Location1 con el Occupant1.

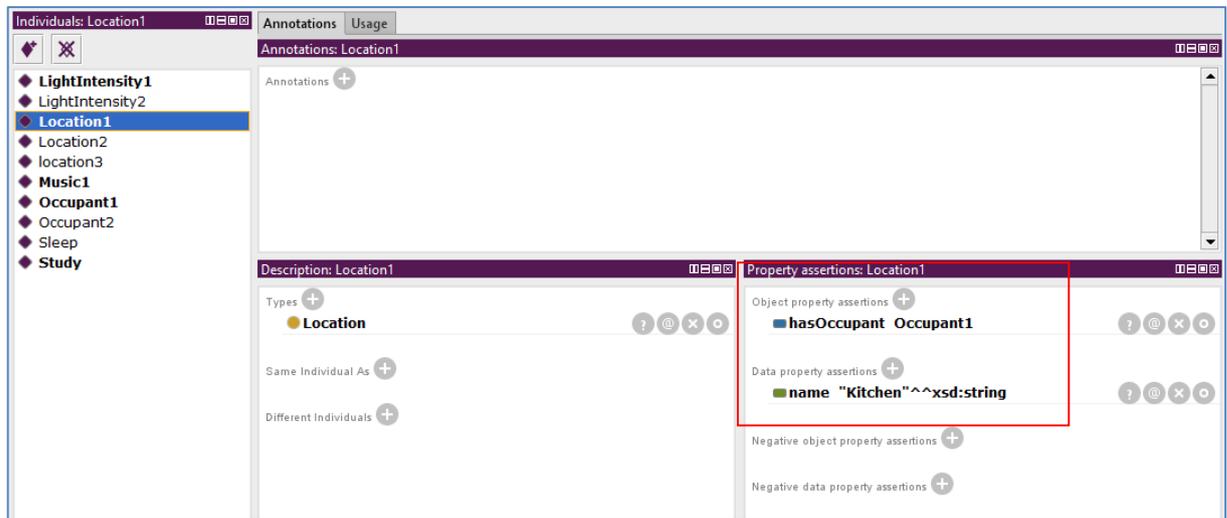


Ilustración 110: Creación Location1

Se crea el tercer individual LightIntensity1 haciendo referencia a su tipo “MediumArtificialIllumination”, agregar su Object property en este caso “mediumArtificialIlluminationIlluminates” hacia Location1, con el propósito de relacionar LightIntensity1 con la Location1.

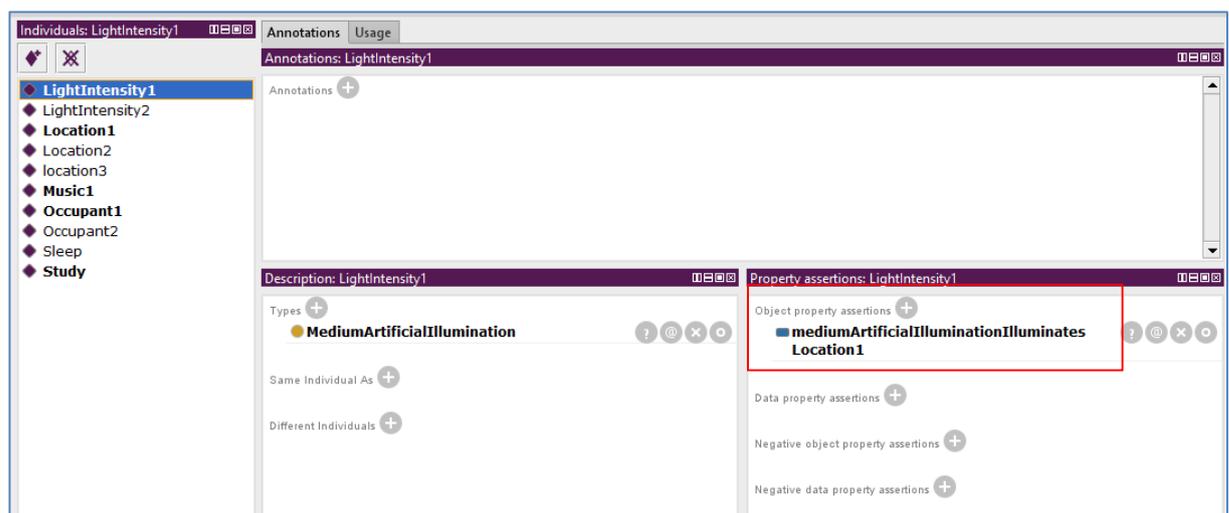


Ilustración 111: Creación LightIntensity1

Se crea el cuarto individual Music1 haciendo referencia a su tipo “MusicAtNight”, agregar su Object property en este caso “hasLight” hacia LightIntensity1, con el propósito de relacionar Music1 con LightIntensity1.

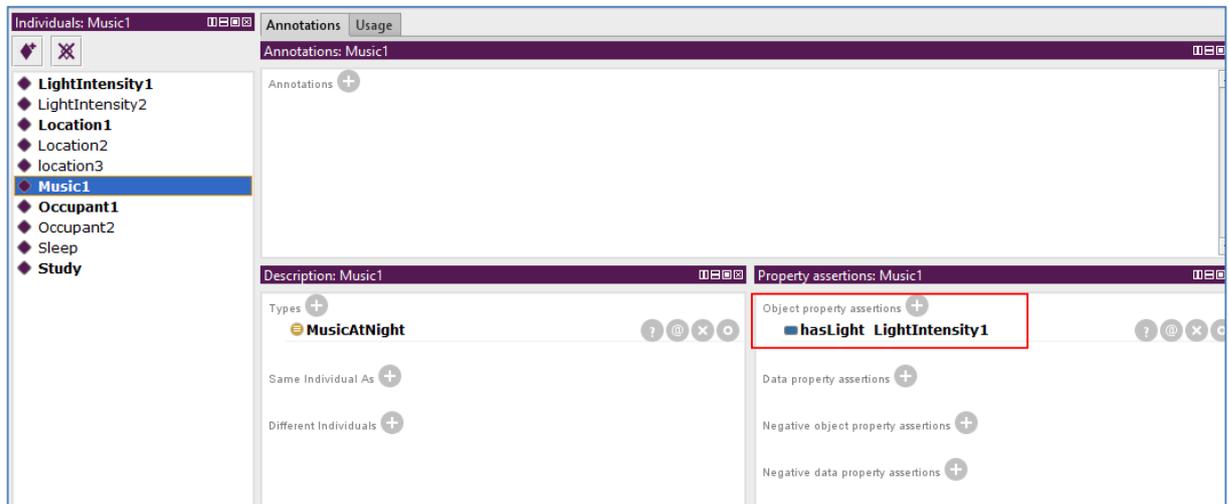


Ilustración 112: Creación Music1

Se crea la Regla de Inferencia para Music1, la ejecución a través del razonador Pellet del mismo Protegé 5.0.

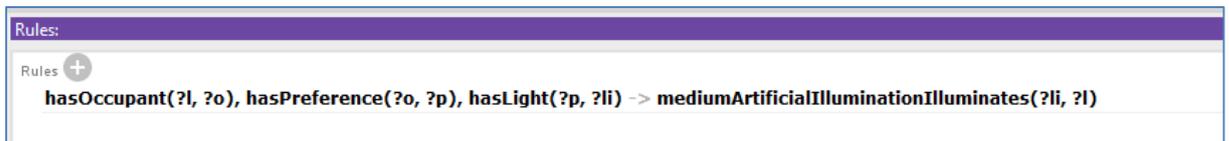


Ilustración 113: Creación Regla de Inferencia Music1

Antes de la ejecución del razonador Pellet no existe resultado que compruebe la regla de inferencia como se muestra en la ilustración siguiente.

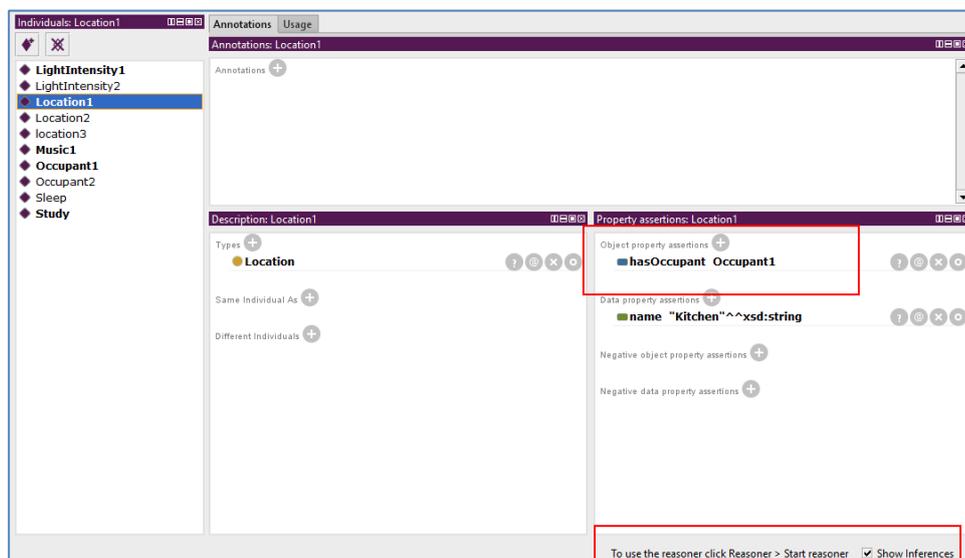


Ilustración 114: Antes ejecución Music1

Después de la ejecución del razonador Pellet se genera el resultado esperado que comprueba la regla de inferencia como se muestra en la siguiente ilustración.

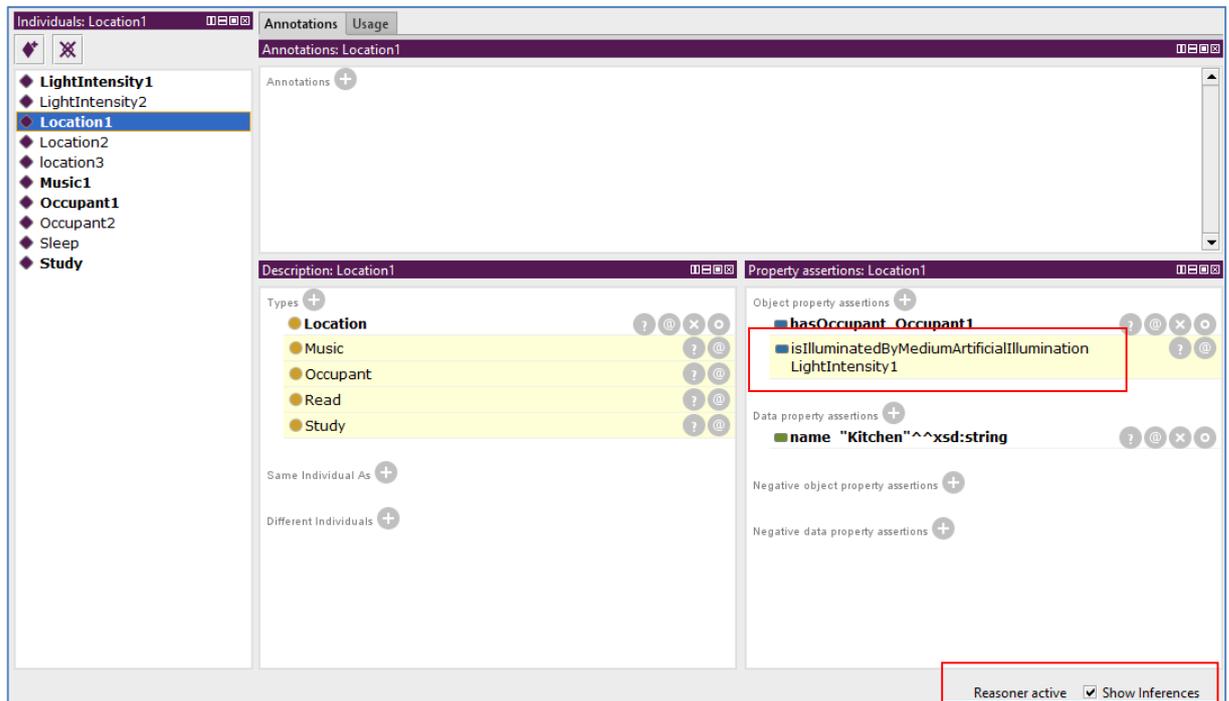


Ilustración 115: Después ejecución Music1

Creación automáticamente de inferencias por las relaciones generadas.

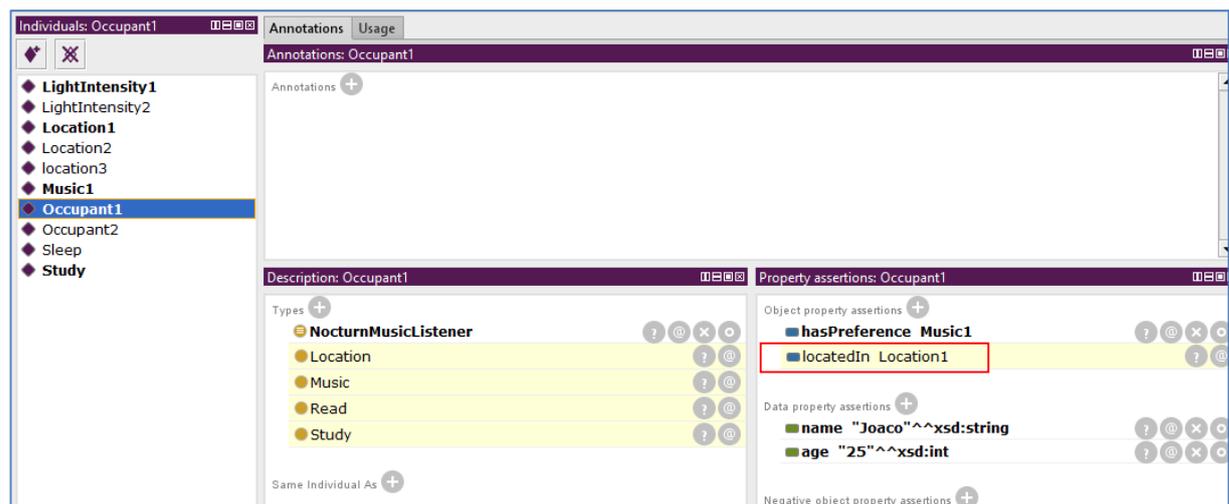


Ilustración 116: Creación de Inferencia Music1

## REGLA DE INFERENCIA N° 2

- **Si Occupant is Sleep Entonces Light NotArtificiallyIlluminatedLocation.**

Se crea el primer individual Occupant2 haciendo referencia a su tipo “Occupant”, los atributos name y age, además de agregar su Object property en este caso “hasPreference” hacia Sleep, con el propósito de relacionar al Occupant2 con la preferencia de Sleep.

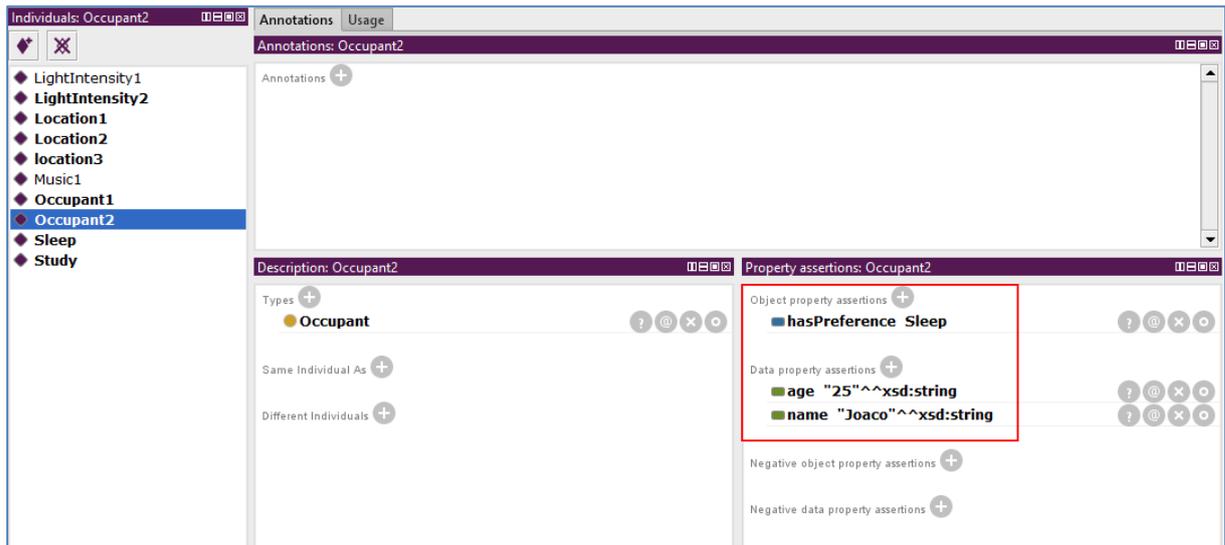


Ilustración 117: Creación Occupant2

Se crea el segundo individual Location2 haciendo referencia a su tipo “Location”, los atributos name, además de agregar su Object property en este caso “hasOccupant” hacia Occupant2, con el propósito de relacionar la location2 con el occupant2.

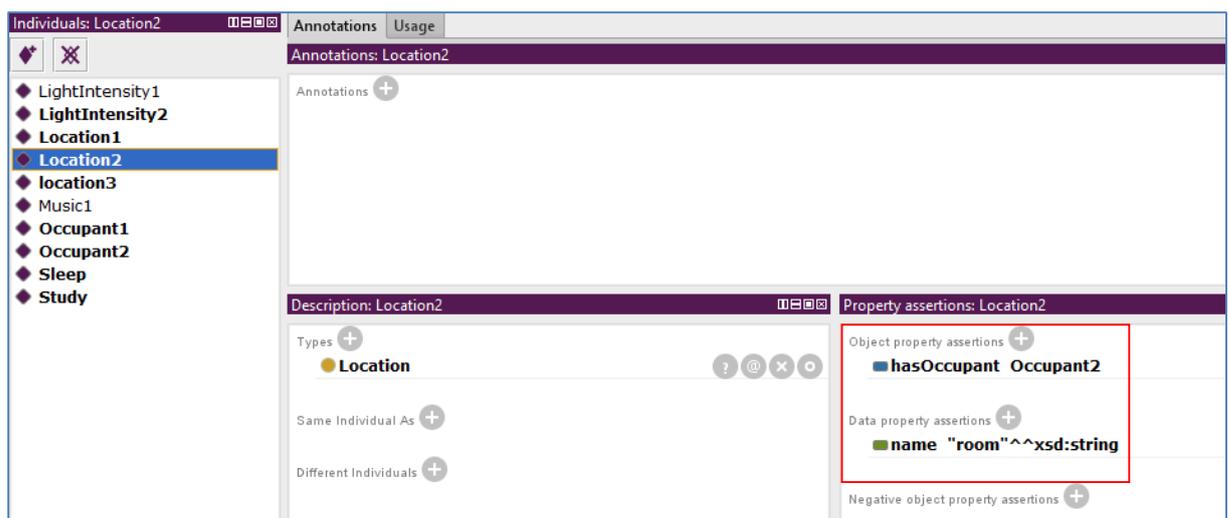


Ilustración 118: Creación Location2

Se crea el tercer individual `LightIntensity2` haciendo referencia a su tipo “`NotArtificialIllumination`”, agregar su Object property en este caso “`noArtificialIlluminationIlluminates`” hacia `Location2`, con el propósito de relacionar `LightIntensity2` con la `Location2`.

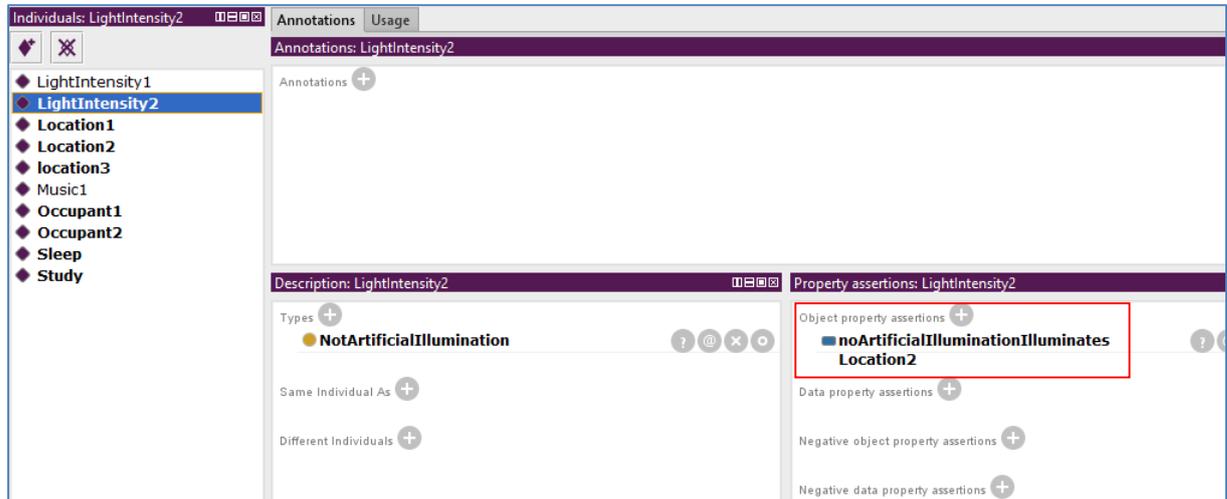


Ilustración 119: Creación `LightIntensity2`

Se crea el cuarto individual `Sleep` haciendo referencia a su tipo “`Sleep`”, agregar su Object property en este caso “`hasLight`” hacia `LightIntensity2`, con el propósito de relacionar `Sleep` con `LightIntensity2`.

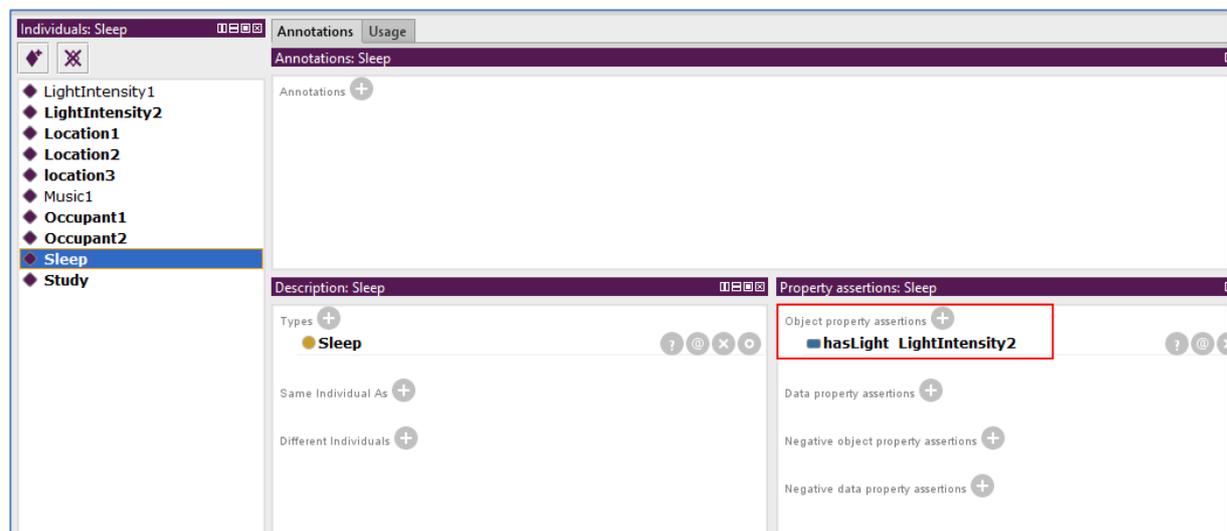


Ilustración 120: Creación `Sleep`

Se crea la Regla de Inferencia para Sleep, la ejecución a través del razonador Pellet del mismo Protegé 5.0.



Ilustración 121: Creación Regla de Inferencia Sleep

Antes de la ejecución del razonador Pellet no existe resultado que compruebe la regla de inferencia como se muestra en la ilustración siguiente.

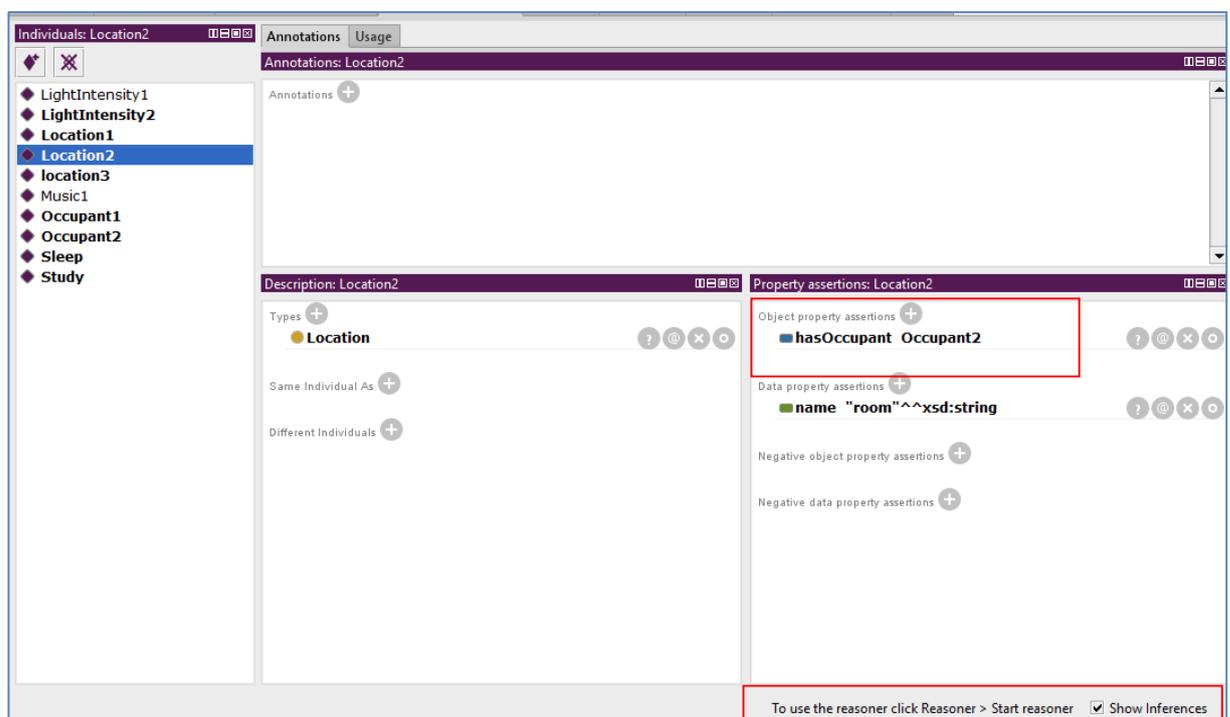


Ilustración 122: Antes de la ejecución Sleep

Después de la ejecución del razonador Pellet se genera el resultado esperado que comprueba la regla de inferencia como se muestra en la siguiente ilustración.

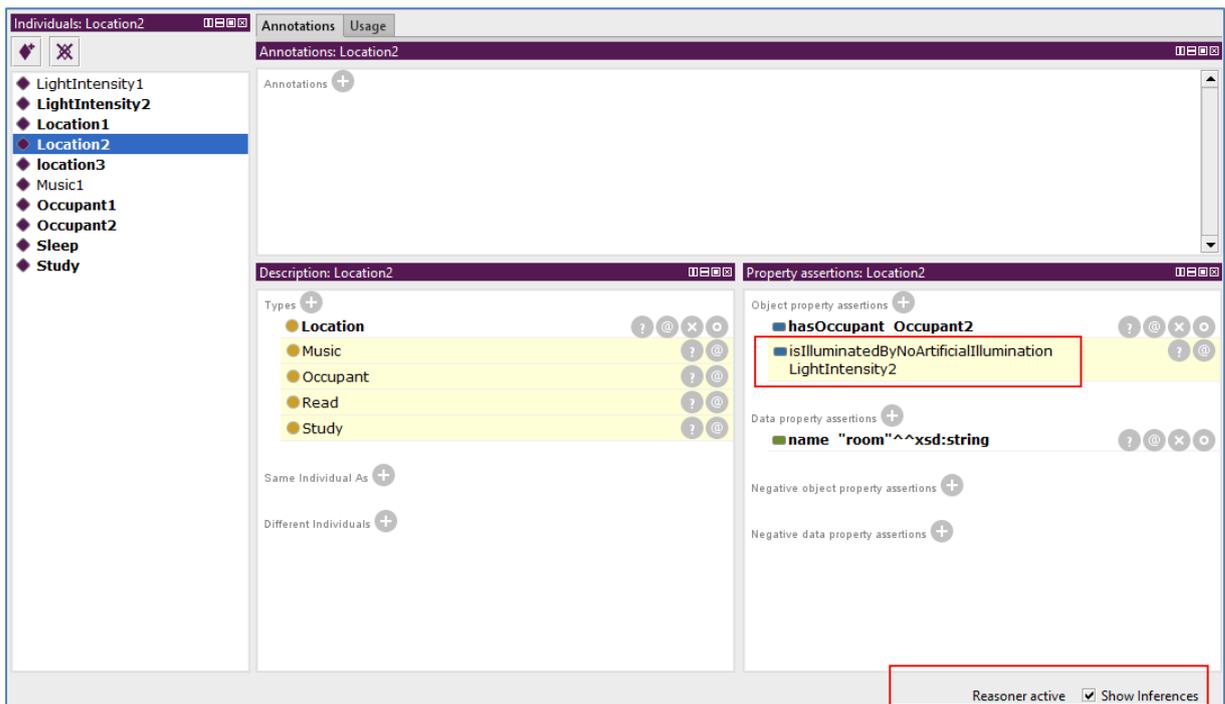


Ilustración 123: Después ejecución Sleep

Creación automáticamente de inferencias por las relaciones generadas.

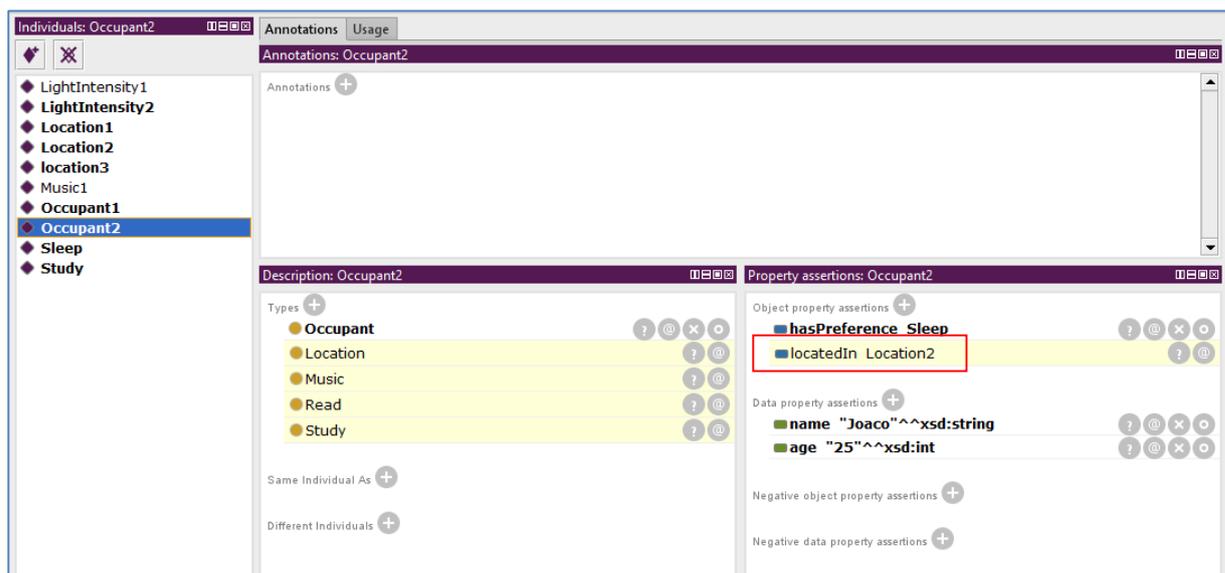


Ilustración 124: Creación de inferencias Sleep

### REGLA DE INFERENCIA N° 3

- **Si Occupant is ReadAtNight Entonces Light HighArtificialIllumination.**

Se crea el primer individual Occupant3 haciendo referencia a su tipo “NocturnReader”, los atributos name y age, además de agregar su Object property en este caso “hasPreference” hacia Read, con el propósito de relacionar al Occupant3 con la preferencia de Read.

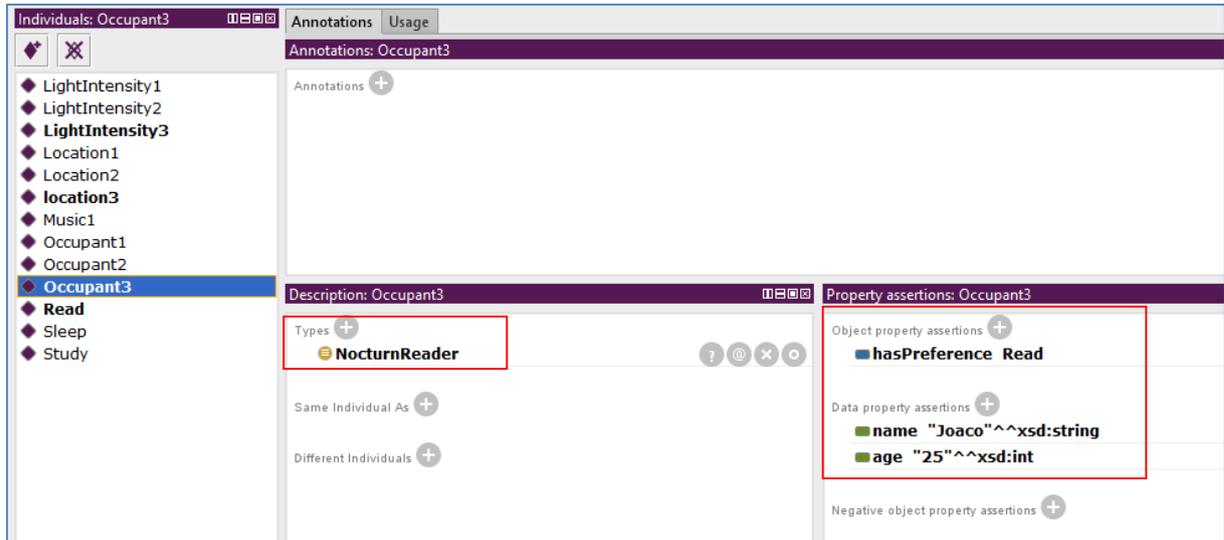


Ilustración 125: Creación Occupant3

Se crea el segundo individual Location3 haciendo referencia a su tipo “Location”, los atributos name, además de agregar su Object property en este caso “hasOccupant” hacia Occupant3, con el propósito de relacionar la Location3 con el Occupant3.

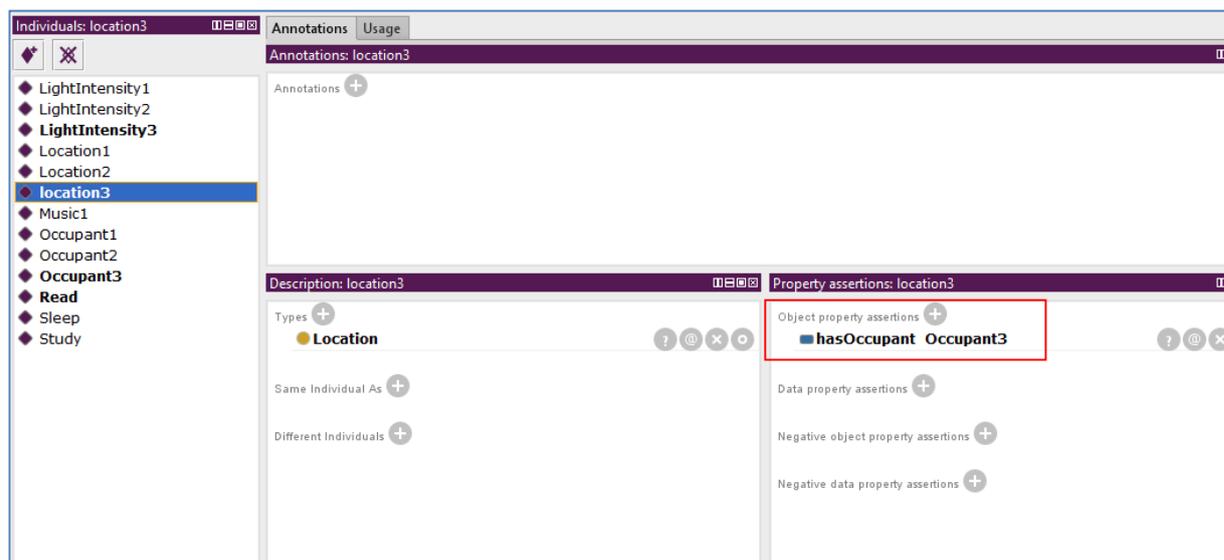


Ilustración 126: Creación Location3

Se crea el tercer individual LightIntensity3 haciendo referencia a su tipo “HighArtificialIllumination”, agregar su Object property en este caso “highArtificialIlluminationIlluminates” hacia Location3, con el propósito de relacionar LightIntensity3 con la Location3.

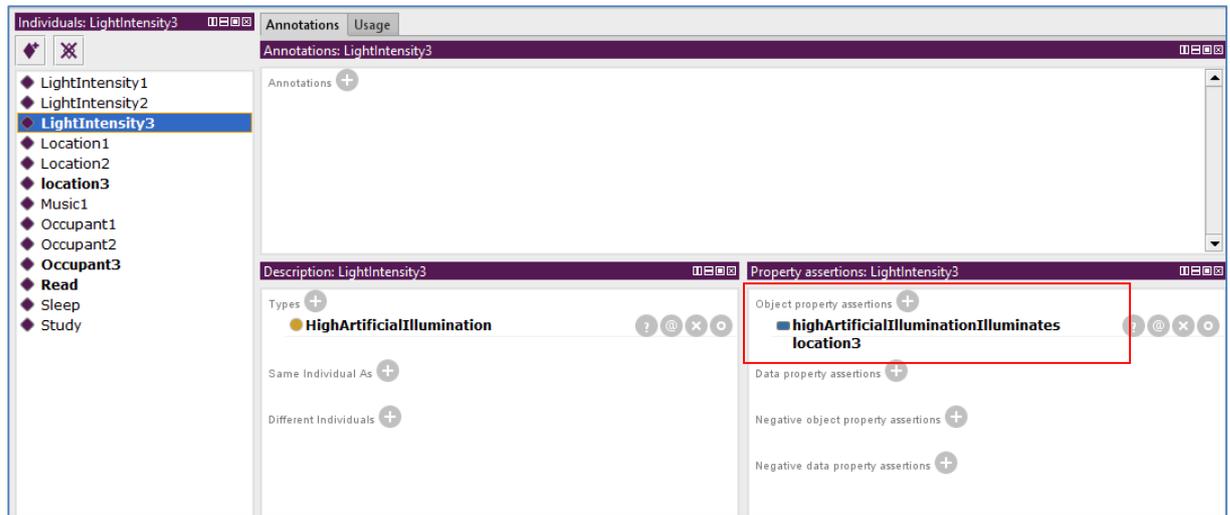


Ilustración 127: Creación LightIntensity3

Se crea el cuarto individual Read haciendo referencia a su tipo “ReadAtNight”, agregar su Object property en este caso “hasLight” hacia LightIntensity3, con el propósito de relacionar Read con LightIntensity3.

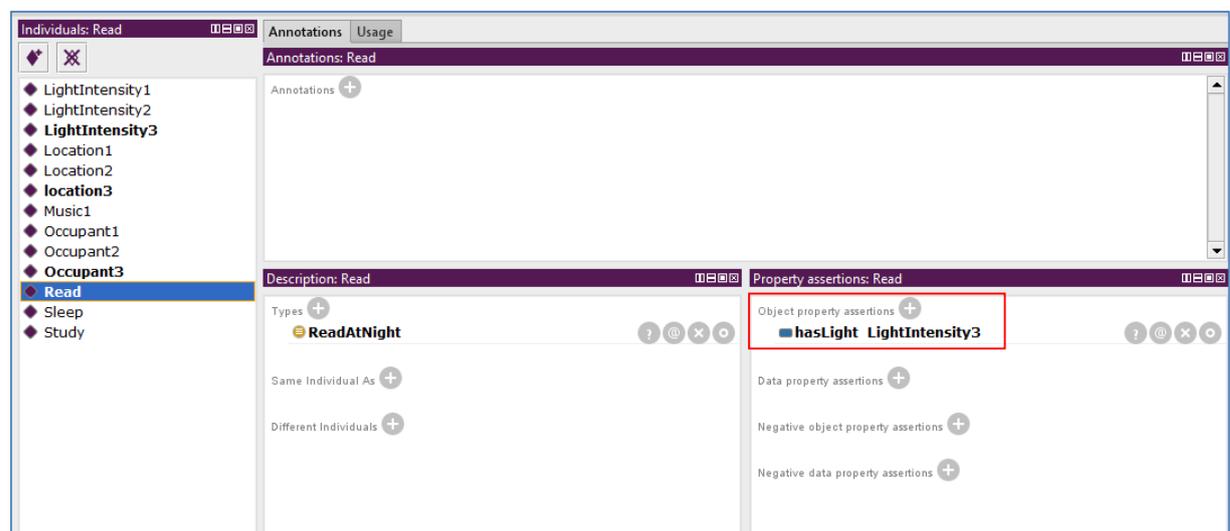


Ilustración 128: Creación Read

Se crea la Regla de Inferencia para Read, la ejecución a través del rasonador Pellet del mismo Protegé 5.0.

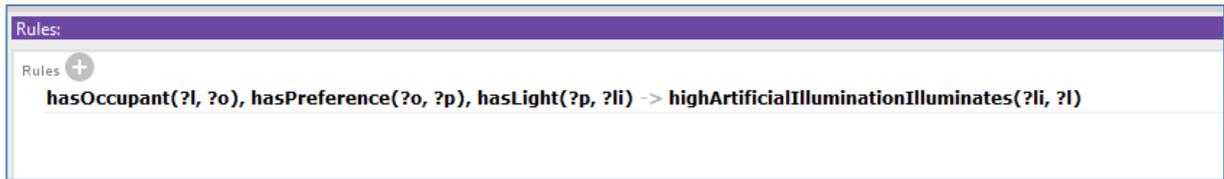


Ilustración 129: Creación Regla de Inferencia Read

Antes de la ejecución del rasonador Pellet no existe resultado que compruebe la regla de inferencia como se muestra en la ilustración siguiente.

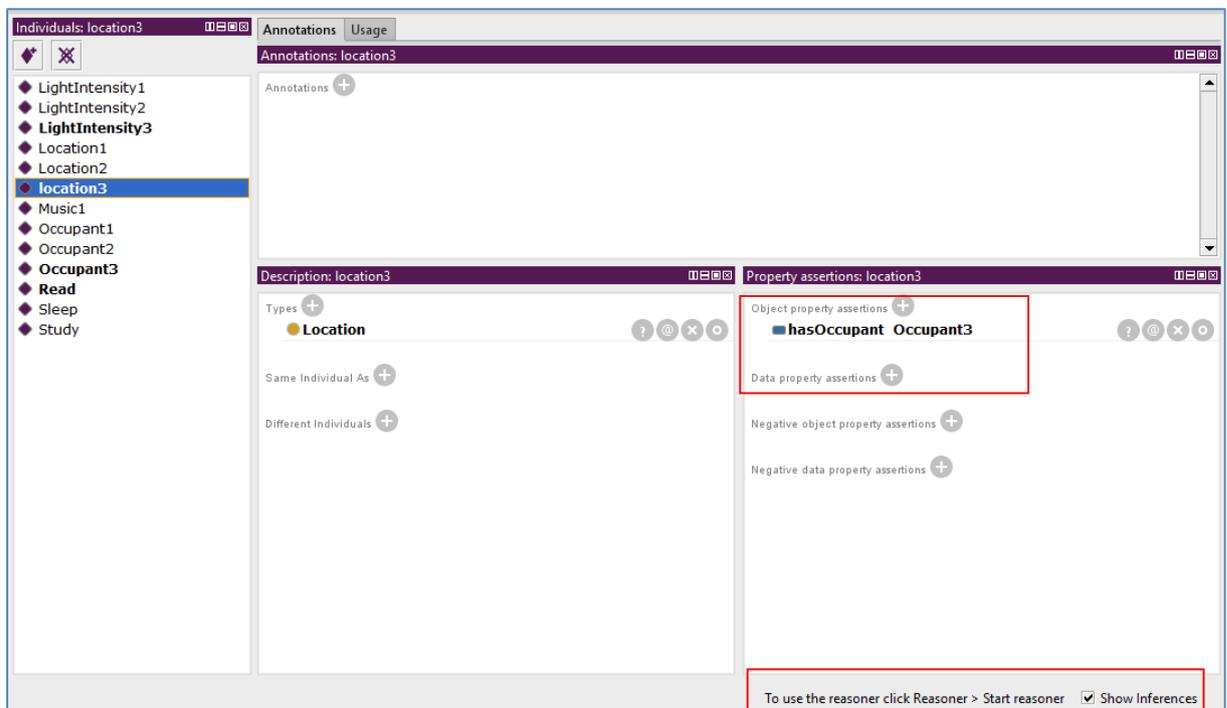


Ilustración 130: Antes ejecución Read

Después de la ejecución del razonador Pellet se genera el resultado esperado que comprueba la regla de inferencia como se muestra en la siguiente ilustración.

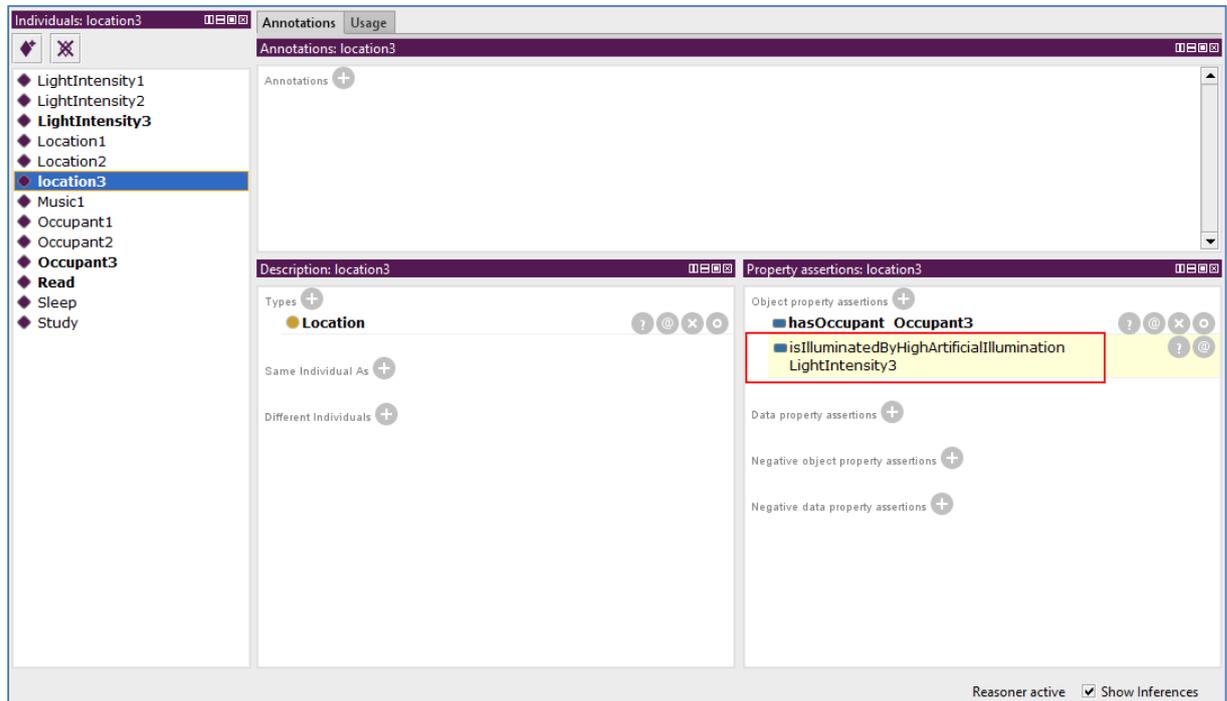


Ilustración 131: Después ejecución Read

Creación automáticamente de inferencias por las relaciones generadas.

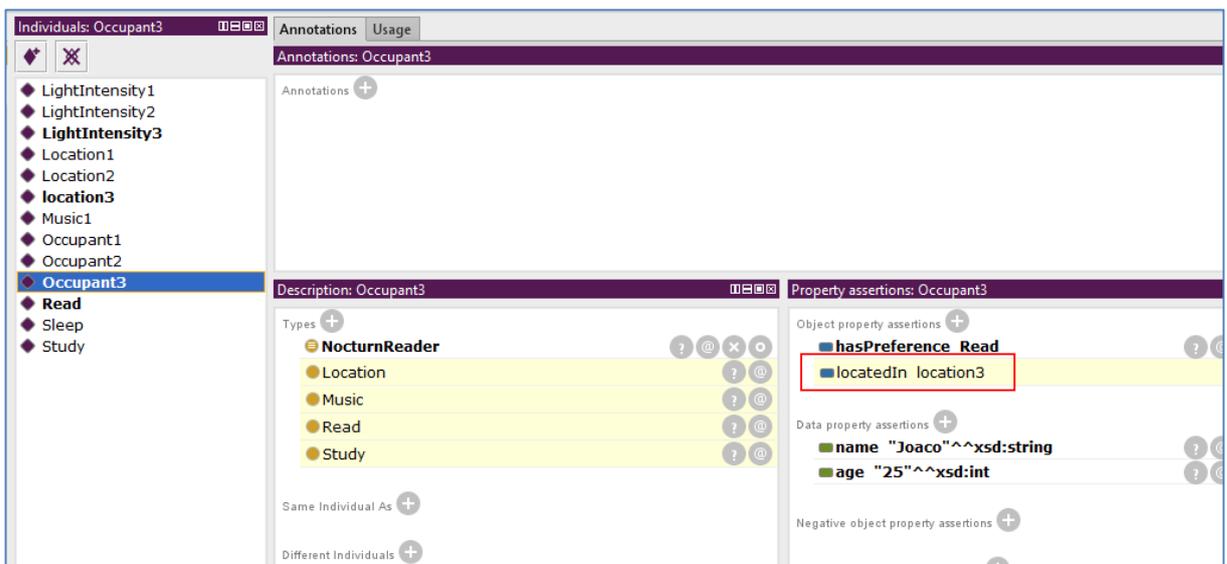


Ilustración 132: Creación de inferencias Read

## REGLA DE INFERENCIA N° 4

- **Si Occupant is StudyAtNight Entonces Light HighArtificialIllumination.**

Se crea el primer individual Occupant4 haciendo referencia a su tipo “NocturnStudent”, los atributos name y age, además de agregar su Object property en este caso “hasPreference” hacia Study, con el propósito de relacionar al Occupant4 con la preferencia de Study.

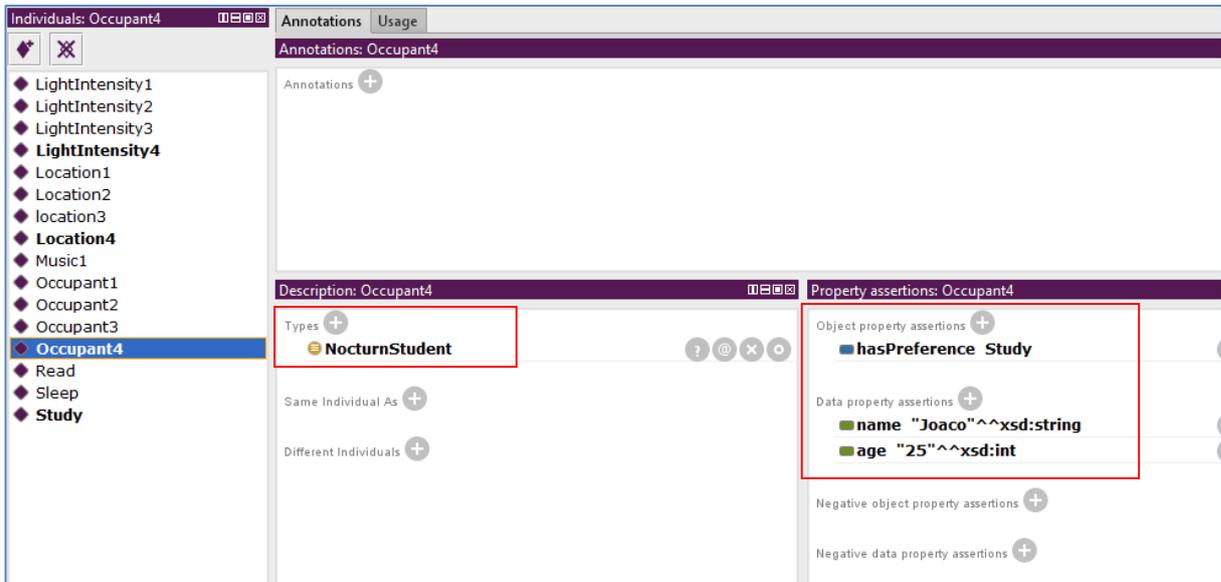


Ilustración 133: Creación Occupant4

Se crea el segundo individual Location4 haciendo referencia a su tipo “Location”, los atributos name, además de agregar su Object property en este caso “hasOccupant” hacia Occupant4, con el propósito de relacionar la Location4 con el Occupant4.



Ilustración 134: Creación Location4

Se crea el tercer individual LightIntensity4 haciendo referencia a su tipo “HighArtificialIllumination”, agregar su Object property en este caso “highArtificialIlluminationIlluminates” hacia Location4, con el propósito de relacionar LightIntensity4 con la Location4.

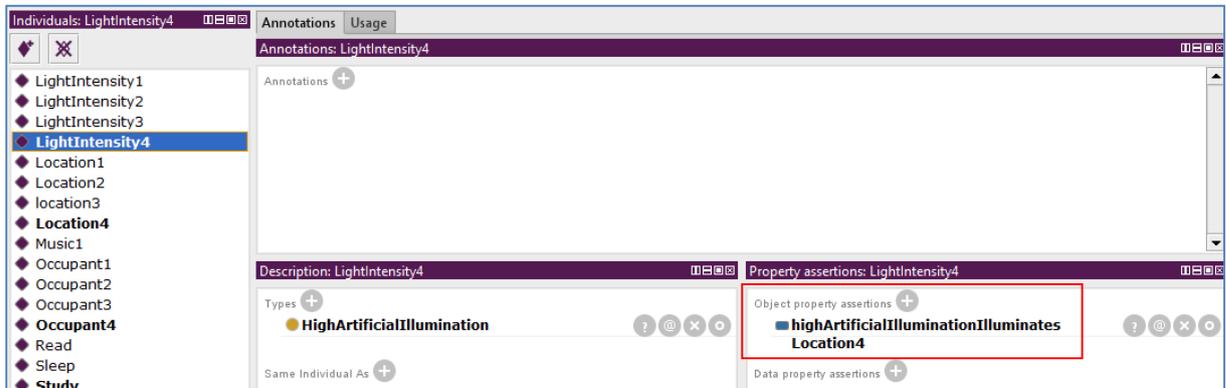


Ilustración 135: Creación LightIntensity4

Se crea el cuarto individual Study haciendo referencia a su tipo “StudyAtNight”, agregar su Object property en este caso “hasLight” hacia LightIntensity4, con el propósito de relacionar Study con LightIntensity4.

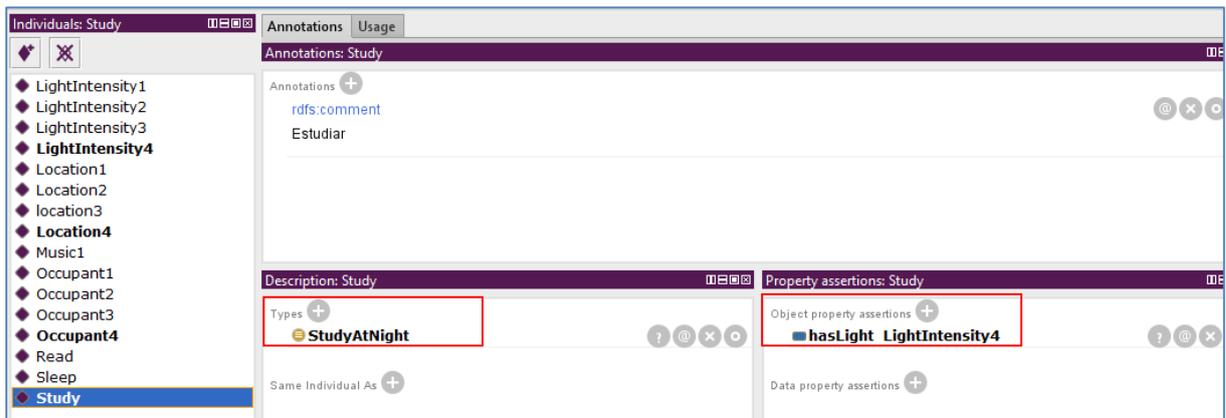


Ilustración 136: Creación Study

Se crea la Regla de Inferencia para Study, la ejecución a través del rasonador Pellet del mismo Protégé 5.0.

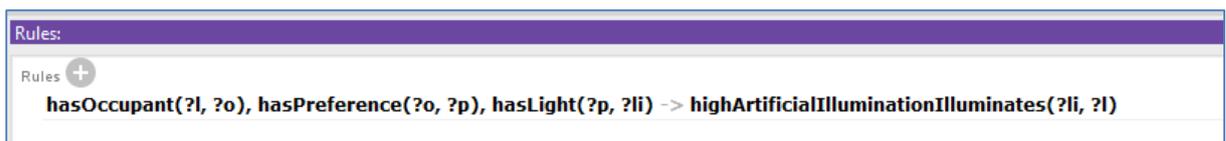


Ilustración 137: Creación Regla de Inferencia Study

Antes de la ejecución del razonador Pellet no existe resultado que compruebe la regla de inferencia como se muestra en la ilustración siguiente.

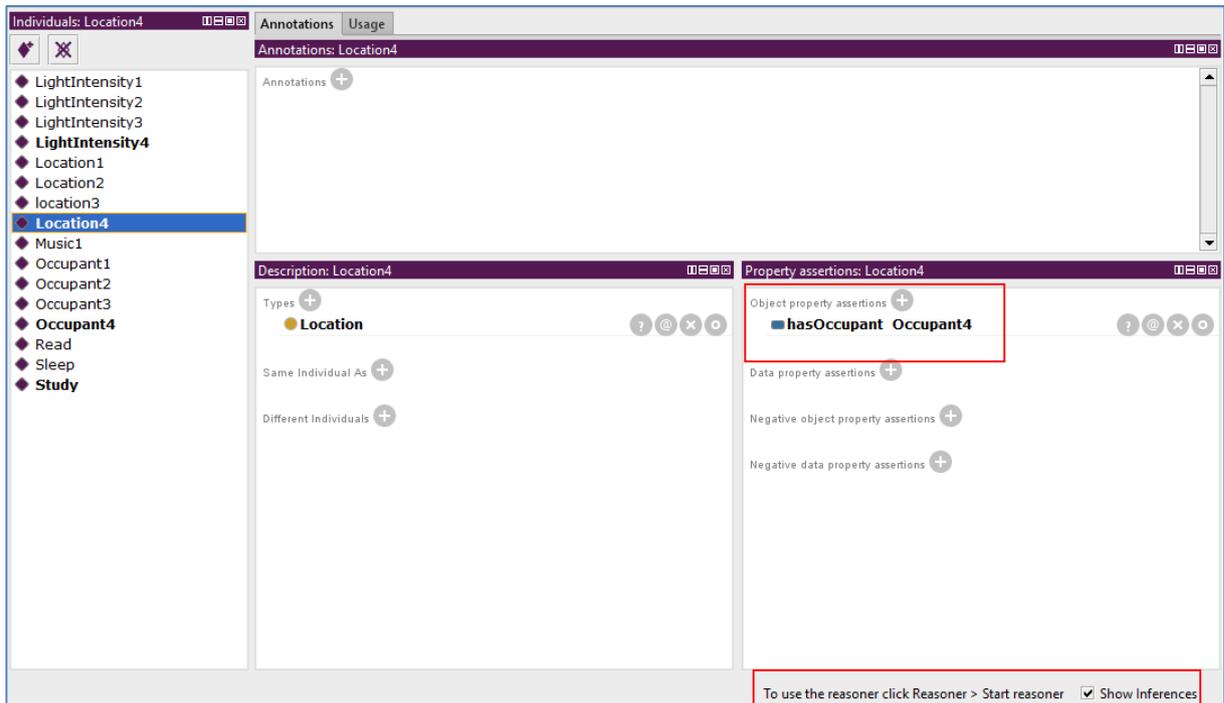


Ilustración 138: Antes ejecución Study

Después de la ejecución del razonador Pellet se genera el resultado esperado que comprueba la regla de inferencia como se muestra en la siguiente ilustración.

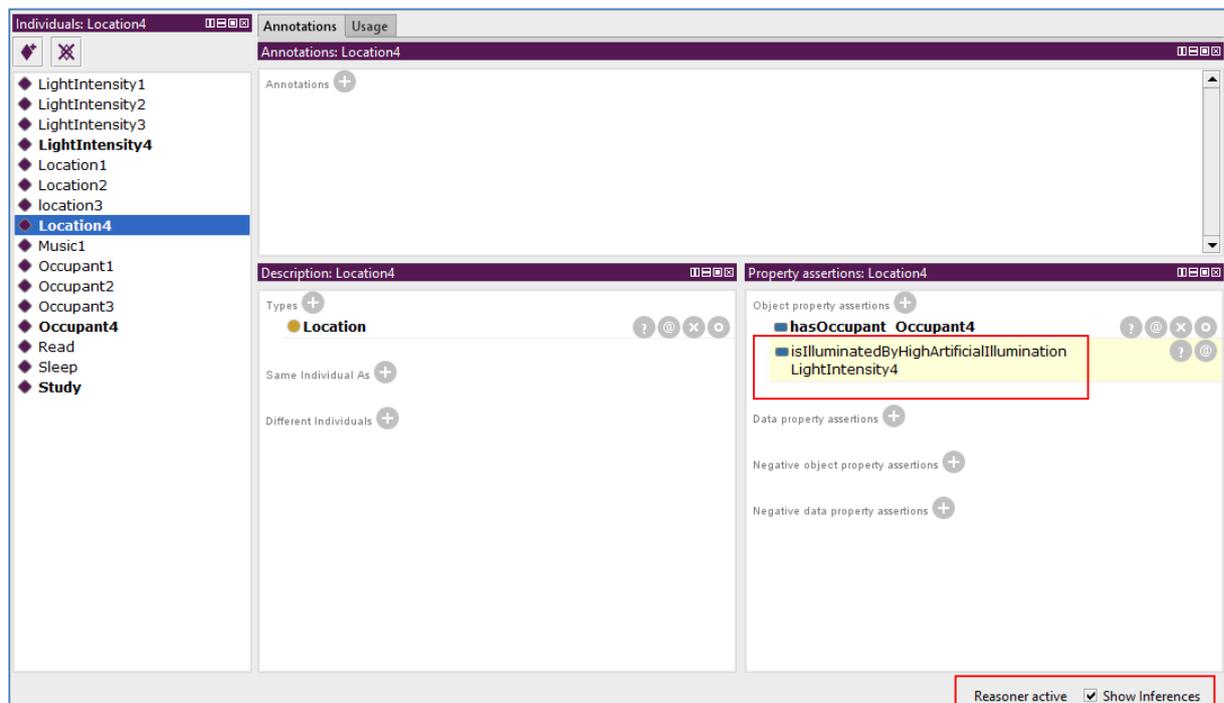


Ilustración 139: Después ejecución Study

Creación automáticamente de inferencias por las relaciones generadas.

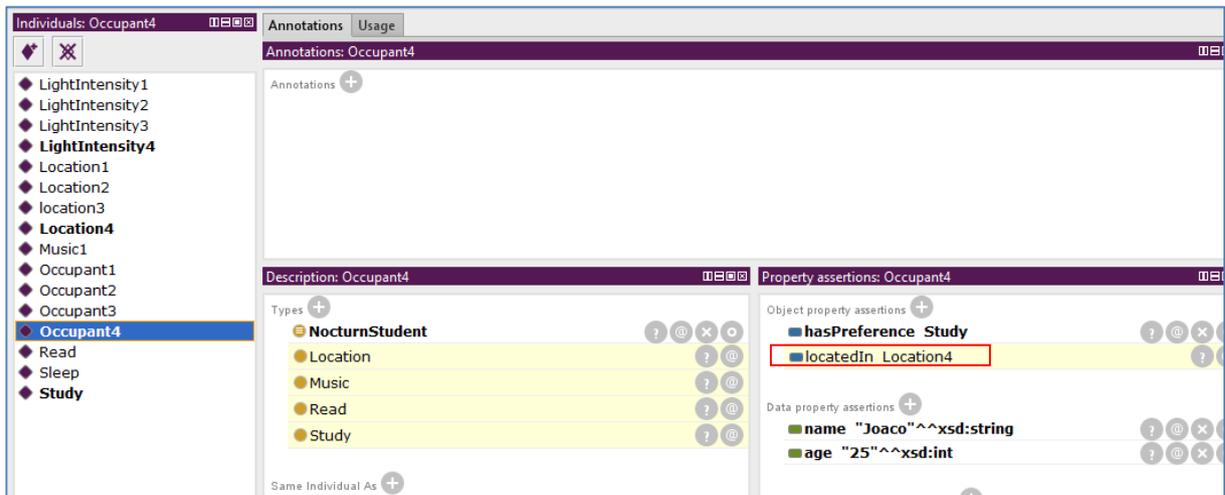


Ilustración 140: Creación regla de inferencia Study

## REGLAS DE INFERENCIA EN PROTÉGÉ

R1: Si Occupant is MusicAtNight Entonces Light MediumArtificialIllumination.

**Rule:** hasOccupant(?l, ?o), hasPreference(?o, ?p), hasLight(?p, ?li) ->  
mediumArtificialIlluminationIlluminates(?li, ?l)

R2: Si Occupant is Sleep Entonces Light NotArtificiallyIlluminatedLocation.

**Rule:** hasOccupant(?l, ?o), hasPreference(?o, ?p), hasLight(?p, ?li) ->  
noArtificialIlluminationIlluminates(?li, ?l)

R3: Si Occupant is ReadAtNight Entonces Light HighArtificialIllumination.

**Rule:** hasOccupant(?l, ?o), hasPreference(?o, ?p), hasLight(?p, ?li) ->  
highArtificialIlluminationIlluminates(?li, ?l)

R4: Si Occupant is StudyAtNight Entonces Light HighArtificialIllumination.

**Rule:** hasOccupant(?l, ?o), hasPreference(?o, ?p), hasLight(?p, ?li) ->  
highArtificialIlluminationIlluminates(?li, ?l)

Rules:	
Rules	+
hasOccupant(?l, ?o), hasPreference(?o, ?p), hasLight(?p, ?li) ->	noArtificialIlluminationIlluminates(?li, ?l)
hasOccupant(?l, ?o), hasPreference(?o, ?p), hasLight(?p, ?li) ->	highArtificialIlluminationIlluminates(?li, ?l)
hasOccupant(?l, ?o), hasPreference(?o, ?p), hasLight(?p, ?li) ->	mediumArtificialIlluminationIlluminates(?li, ?l)

Ilustración 141: Reglas de Inferencia

Realizando la contrastación de la hipótesis

### 5.1 Indicador de Número de Inferencias de la iluminación

Se mide el número de inferencias acertadas que resulta después de la ejecución de las cuatro reglas de inferencia veinte veces cada una.

- ✓ Pre-Test (T1): Medición previa de la variable dependiente a ser utilizada.
- ✓ Post-Test (T2): Corresponde a la nueva medición de la variable dependiente a ser utilizada.

Dónde: **T1\_X\_T2**

**T1:** Número de inferencias acertadas antes de la ontología.

**X:** (Aplicación de la variable independiente) **Ontología de Patrón de Comportamiento.**

**T2:** Número de inferencias acertadas luego de la aplicación de 4 reglas de inferencias de la ontología.

A continuación, se muestran las mediciones hechas a la variable correspondiente:

Dónde:

**X<sub>Ai</sub>:** Número de inferencias acertadas antes de la ontología.

**X<sub>Di</sub>:** Número de inferencias acertadas con la aplicación de 4 reglas de inferencias después del desarrollo de la ontología.

Tabla 8: Comparación de Número de Inferencias Acertadas

Número de inferencias de la Iluminación.	Cantidad Ejecutadas	Inferencias Acertadas	
		X <sub>Ai</sub>	X <sub>Di</sub>
MusicAtNight	20	0	15
Sleep	20	0	18
ReadAtNight	20	0	14
StudyAtNight	20	0	19
<b>TOTALES</b>	<b>80</b>	<b>0</b>	<b>66</b>

Tabla 9: Fuente de Datos Históricos – Indicador de Inferencias Acertadas

t-Test: Paired Two Sample for Means

	Variable 1	Variable 2
Mean	0	16.5
Variance	0	5.666666667
Observations	4	4
Pearson Correlation	0	
Hypothesized Mean Difference	0	
Df	3	
<b>t Stat</b>	<b>-13.86277283</b>	
P(T<=t) one-tail	0.000406269	
t Critical one-tail	2.353363435	
<b>P(T&lt;=t) two-tail</b>	<b>0.000812539</b>	
t Critical two-tail	3.182446305	

### Hipótesis Estadística:

H1: Una Ontología de patrón de comportamiento como controlador Web permitirá anticiparse a las preferencias de los usuarios en el departamento domotizado 301 del proyecto Residencial Los Jazmines de la empresa PROARCO S.A.C. en Trujillo el año 2016.

H2: Una Ontología de patrón de comportamiento como controlador Web no permitirá anticiparse a las preferencias de los usuarios en el departamento domotizado 301 del proyecto Residencial Los Jazmines de la empresa PROARCO S.A.C. en Trujillo el año 2016.

En cuanto a los resultados obtenidos en la parte estadística, el valor de P es menor al nivel de significancia 0.09, por lo cual se acepta la Hipótesis H1, por lo que el número de inferencias acertadas **es mayor después del desarrollo de la Ontología de patrón de Comportamiento.**

Por otro lado, mediante la ejecución del razonador Pellet sobre la ontología como se muestra en el capítulo V a través de las ilustraciones 109 – 140 donde se ponen a prueba cuatro reglas de inferencia sobre lumínica, nos permite aseverar que sí es posible anticiparse a las preferencias de los usuarios de un departamento domotizado.

## CAPITULO VI: CONCLUSIONES

Presentamos en este capítulo las conclusiones obtenidas a raíz de este proyecto, las cuales basadas por nuestros objetivos específicos concluimos lo siguiente:

- En el análisis de glosario de términos y taxonomía de conceptos, se identificaron un total de 76 conceptos, reutilizando algunos de ontologías tales como: DogOnt, Energy Profile Ontology, DogPower y CONON. Finalmente se elaboró el diagrama de la taxonomía.
- En la definición de relaciones binarias, se identificaron 28 relaciones binarias de la ontología, así como sus características más relevantes, 26 Asimétrica, 26 Irreflexivas y 5 Funcionales
- Se determinaron 18 atributos, 4 axiomas y 4 reglas de inferencia de la ontología.
- Se implementó la Ontología en el software Protégé y razonador Pellet generando inferencia en 4 de las reglas, debido a que al considerar una actividad que se repite con frecuencia (leer, escuchar música, dormir) se enciende la luminaria en el nivel requerido si no hay iluminación natural, demostrando que la ontología puede anticiparse a las preferencias de los usuarios para la configuración de la iluminación en un departamento domotizado.

## CAPITULO VIII: RECOMENDACIONES

Este capítulo final comprende las recomendaciones, en caso que se desee dar continuidad al Proyecto Ontología de Patrón de Comportamiento, debe tener en cuenta, los siguientes aspectos:

- Se sugiere expandir la simulación incluyendo los tipos de ocupantes (papá, mamá e hijos) diferenciando a cada uno con su respectivo comportamiento.
- Se recomienda formular más reglas de inferencia para la ontología propuesta de tal manera que esta vea su dinamismo incrementado y las consultas generen unas inferencias más profundas.
- Ahondar en esta rama de la tecnología para seguir contribuyendo con la generación de conocimiento para así expandir su aplicación a otros ámbitos.
- Expandir el módulo de la ontología propuesta de tal manera que no solo abarque el proyecto de iluminación del departamento, sino que además el control de los dispositivos, sistema de seguridad, etc.
- Utilizar el software Protégé versión 5, para el desarrollo de la ontología, ya que ésta es la más completa comparándola con las versiones 3.4 y 4, las cuales necesitan de la instalación de los razonadores, independientemente de la del software principal (Protégé).

## REFERENCIAS BIBLIOGRÁFICAS

- Haughey, D. (s.f.). *Matriz RACI*. Recuperado el 14 de setiembre de 2015, de ProjectSmart.co.uk:  
<https://www.projectsmart.co.uk/raci-matrix.php>
- ABC, D. (s.f.). Definición de Domótica.
- ABC, D. (s.f.). Definición de Ontología.
- AutoresTesis. (2016).
- Bautista-Zambrana, M. R. (2015). ScienceDirect. En M. R. Bautista-Zambrana, *Methodologies to build ontologies for terminological purposes* (págs. 264-269).
- Boletín DINTEL. (01 de Marzo de 2011). *Tecnocom renueva los medios de pago del principal banco de Colombia*. Recuperado el 05 de Septiembre de 2012, de  
<http://www.boletindintel.es/BoletinesAyS/Publico/PresentaContenido.php?Fase=1%20&%20Referencia=1618>
- Bonino, D. &. (2014). Exploiting semantic technologies in smart environments and grids: Emerging roles and case studies. . En *Science of Computer Programming* (págs. 112–134).  
Obtenido de <http://dx.doi.org/10.1016/j.scico.2014.02.018>
- Bonino, D., & Corno, F. (2008). *Dogont-Ontology Modeling For Intelligent Domotic Environments. The Semantic Web-Iswc 2008. Springer Berlin Heidelberg P. 790-803*.
- Bonino, D., Corno, F., & Razzak, F. (2011). Enabling machine understandable exchange of energy consumption information in intelligent domotic environments. *Energy and Buildings*(43), 1392–1402.
- Camazón, J. N. (2011). *Sistemas operativos monopuesto*. Editex.
- Carbonell, S. M. (2014). *"Interfaz para el Control de un Entorno Doméstico Mediante Voz"*.
- Castro, G. D. (2014). Recuperado el 07 de Octubre de 2015, de  
<https://www.dspace.espol.edu.ec/bitstream/123456789/25068/1/TESIS%20ICM-Control%20y%20operaci%C3%B3n%20log%C3%ADstica.pdf>
- Chamochumbi, D. D. (18 de Marzo de 2013). *Diagnóstico, Análisis y Propuesta de Mejora al Proceso de Gestión de Interrupciones imprevistas en el Suministro Eléctrico de Baja Tensión*. Obtenido de  
[http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/4791/SALAS\\_CHAMOCHUMBI\\_DANIEL\\_DIAGNOSTICO\\_ELECTRICIDAD.pdf?sequence=1](http://tesis.pucp.edu.pe/repositorio/bitstream/handle/123456789/4791/SALAS_CHAMOCHUMBI_DANIEL_DIAGNOSTICO_ELECTRICIDAD.pdf?sequence=1)
- Chen, H. P. (2004). In Mobile and Ubiquitous Systems: Networking and Services. MOBIQUITOUS The First Annual International IEEE. En *Soupa: Standard ontology for ubiquitous and pervasive applications* (págs. 258-267).
- cmapspublic3. (s.f.). <http://cmapspublic3.ihmc.us/rid=1JCTNSLFS-21QC96K-51/Ontolog%C3%ADa%20-%20CMap%2023-04-2011.cmap>.

- Consumoteca. (04 de Octubre de 2009). *Las Tarjetas Bancarias: Tipos y Diferencias*. Recuperado el 02 de Septiembre de 2012, de <http://www.consumoteca.com/economia-familiar/medios-de-pago/las-tarjetas-bancarias%3A-tipos-y-diferencias>
- Corcho O, F.-L. M.-P. (s.f.). Construcción de ontologías legales con la metodología. Facultad de Informática. Universidad Politécnica de Madrid, España.
- Corona Correa, S. (2012). *Factores críticos de éxito en el proceso de migración de datos*. (D. G. Académico, Productor, & Universidad Nacional Autónoma de México) Recuperado el 30 de Agosto de 2012, de [http://www.paginaspersonales.unam.mx/files/35/Susana\\_Corona\\_Correa\\_Mexico\\_.pdf](http://www.paginaspersonales.unam.mx/files/35/Susana_Corona_Correa_Mexico_.pdf)
- Dario Bonino, F. C. (2010). *"Enabling machine understandable exchange of energy consumption information in intelligent domotic environments"*. Torino, Italy: Politecnico di Torino, Dipartimento di Automatica ed Informatica, Corso Duca degli Abruzzi 24, 10129 .
- Diario El Comercio. (22 de Mayo de 2012). *Consumo con tarjetas de crédito se elevó S/. 14.751 millones en abril*. Recuperado el 29 de Agosto de 2012, de <http://elcomercio.pe/economia/1418034/noticia-consumo-tarjetas-credito-se-elevo14751-millones-abril>
- ECONOMIST INTELLIGENCE UNIT (EIU) . (2004). *"The Economist Intelligence Unit's quality of life index"* . *TheEconomist* . . Obtenido de [http://www.economist.com/media/pdf/QUALITY\\_OF\\_LIFE.pdf](http://www.economist.com/media/pdf/QUALITY_OF_LIFE.pdf)
- EcuRed.com*. (14 de Diciembre de 2010). Recuperado el viernes, 28 de Setiembre de 2015, de ECURED- Conocimiento con todos y para todos: [http://www.ecured.cu/index.php/Sistema\\_inform%C3%A1tico](http://www.ecured.cu/index.php/Sistema_inform%C3%A1tico)
- El Comercio. (29 de Enero de 2016). *¿Qué pagamos cuando pagamos el recibo de electricidad?* Obtenido de <http://elcomercio.pe/economia/negocios/que-pagamos-cuando-pagamos-recibo-electricidad-noticia-1874830>
- Facultad de Ingeniería - Universidad Nacional de la Plata. (s.f.). *Qué es un Procedimiento*. Recuperado el 26 de 09 de 2012, de <http://www.ing.unlp.edu.ar/produccion/introing/bib/Que%20es%20un%20procedimiento.pdf>
- Frain, B. (2015). *Responsive Web Design with HTML5 and CSS3* (Second Edition ed.). Packt Publishing Ltd.
- Garcia , E. (14 de febrero de 2015). *Herramientas para la gestion de tareas*. Obtenido de lancetalent.com: <http://www.lancetalent.com/blog/las-10-mejores-herramientas-para-la-gestion-de-tareas/>
- González, M. S. (s.f.). Patrones de comportamiento. Miami.
- Hrenhata. (s.f.). *Cómo ahorrar energía instalando mecanismos de control eficiente de la iluminación*. Obtenido de <http://www.five.es/calidadentuvivienda/ahorrar-energia-en-tu-vivienda?id=68#> - ¿cómo-puedo-solucionarlo

<https://en.wikipedia.org/wiki/SPARQL>. (s.f.).

Hudicka, J. (Abril de 1998). *Overview Data Migration Methodology*. Recuperado el 02 de Septiembre de 2012, de [http://www.dulcian.com/articles/overview\\_data\\_migration\\_methodology.htm](http://www.dulcian.com/articles/overview_data_migration_methodology.htm)

IBM Corporation. (2010). *Introduction to DB2 for z/OS*. Recuperado el 29 de Agosto de 2012, de [http://publib.boulder.ibm.com/infocenter/zos/basics/index.jsp?topic=/com.ibm.db2z.doc.intro/db2z\\_zbsoverviewdb2.html](http://publib.boulder.ibm.com/infocenter/zos/basics/index.jsp?topic=/com.ibm.db2z.doc.intro/db2z_zbsoverviewdb2.html)

IBM Corporation. (2010). *z/OS Concepts - What is TSO*. Recuperado el 29 de Agosto de 2010, de [http://publib.boulder.ibm.com/infocenter/zos/basics/index.jsp?topic=/com.ibm.zos.zco.ncepts/zconc\\_whatistso.htm](http://publib.boulder.ibm.com/infocenter/zos/basics/index.jsp?topic=/com.ibm.zos.zco.ncepts/zconc_whatistso.htm)

Informática Corporation. (2012). *Data Migration*. Recuperado el 30 de Agosto de 2012, de <http://www.informatica.com/es/solutions/enterprise-data-integration-and-management/data-migration/>

*Informatica- Hoy*. (2012). Obtenido de [www.informatica-hoy.com](http://www.informatica-hoy.com): <http://www.informatica-hoy.com.ar/aprender-informatica/Que-es-un-sistema-informatico.php>

*Informatica-Hoy*. (2012). *Qué es un sistema informático*. Obtenido de <http://www.informatica-hoy.com>: <http://www.informatica-hoy.com.ar/aprender-informatica/Que-es-un-sistema-informatico.php>

La Industria. (08 de Marzo de 2016). *Venta de departamentos disminuye en más de 60 %*. Obtenido de <http://www.laindustria.pe/actualidad/economia/venta-de-departamentos-disminuye-en-mas-de-60>

López, G., Solano, J., & Cruz, M. (2014). Prototipo de supervisión del control por medio de una aplicación WEB y adquisición de datos de un sistema domótico. *Ciencias de la Ingeniería y Tecnología*, 7, 78-93.

Ministerio de Economía y Finanzas. (27 de 08 de 2013). *Sistema Electrónico de Contrataciones del Estado*. Recuperado el 26 de 09 de 2015, de <http://www2.seace.gob.pe/>

Perales, D. M. (2013). *UNIX a base de ejemplos* (4 ed.). España.

Perú21. (04 de Enero de 2016). *Osinerghin anunció que tarifas eléctricas aumentarán desde hoy*. Obtenido de <http://peru21.pe/economia/osinerghin-anuncio-que-tarifas-electricas-aumentaran-desde-hoy-2235816>

Ramos, L. (2014). *“Uma Proposta de Ontologia para Residências”*.

Red Huánuco. (2011). *Red Huánuco - Soluciones Web Integrales*. Recuperado el 04 de Septiembre de 2012, de <http://www.redhuanuco.com/emarketing.php>

Rodriguez, E. (2006). *Asignación multicriterio de tareas a trabajadores polivalentes*. Obtenido de Tesis UPC: <http://eoli.upc.edu/projects/OTT/2006-ericka-rodriguez-tesi.pdf>

S.A.C, C. d. (Marzo de 2016). Proyecto de Domotización “Residencial Los Jazmines”. (B. J. Br. Stefany Cacho A, Entrevistador)

- Schneider Electric. (Enero de 10). Obtenido de [http://www.schneider-electric.com.ar/documents/solutions/catalogo\\_soluciones.pdf](http://www.schneider-electric.com.ar/documents/solutions/catalogo_soluciones.pdf)
- Significados. (s.f.). <http://www.significados.com/ontologia/>.
- Siles, R., & Mondelo, E. (2012). *PMP- Guia de Gestion de Proyectos*.
- Sommaruga, L., Perri, A., & Furfari, F. D.-E. (2005). "An Ontology For Human Home Interaction. In: *Swap*".
- Tarjeta de Crédito. (2012). *Tarjetas de Crédito*. Recuperado el 05 de Septiembre de 2012, de <http://tarjetadecreditositio.com/tarjeta-de-credito-rapida/>
- Tecnocom. (2011). *SAT - Sistema de Administración de Tarjetas*. Recuperado el 29 de Agosto de 2012, de <http://www.tecnocom.es/Documentos%20Web%20Tecnocom/Doc%20web%20SAT.pdf>
- Universidad de Oviedo. (2010). *ATC-Distribuidas*. (D. d. Informática, Editor) Recuperado el 29 de Agosto de 2012, de Conceptos Básicos: [http://www.atc.uniovi.es/inf\\_superior/atc/DISTRIBUIDAS/01-Conceptos\\_Basicos.pdf](http://www.atc.uniovi.es/inf_superior/atc/DISTRIBUIDAS/01-Conceptos_Basicos.pdf)
- Urrelo, L. V. (2016). "Ontología de Patrón de Comportamiento para Incrementar el Ahorro de Energía Eléctrica Buscando Justificar la Inversión en Departamentos Domotizados". Trujillo.
- W3C. (2006). *Time Ontology in OWL*. Obtenido de <http://www.w3.org/TR/owl-time/>
- Wang, X. H. (2004). Proceedings of the Second IEEE Annual Conference IEEE. En *Ontology based context modeling and reasoning using OWL. In Pervasive Computing and Communications Workshops* (págs. 18- 22).
- Werner J. Vanzyl. (19 de Noviembre de 2010). *Asignación de tareas humanas simultáneas en WebSphere Process Server V7*. Obtenido de [www.ibm.com: http://www.ibm.com/developerworks/ssa/websphere/library/techarticles/1006\\_vanzyl/1006\\_vanzyl.html](http://www.ibm.com/developerworks/ssa/websphere/library/techarticles/1006_vanzyl/1006_vanzyl.html)
- ZDNet. (07 de Septiembre de 2011). *Mainframe Ubuntu Linux?* Recuperado el 05 de Septiembre de 2012, de <http://www.zdnet.com/blog/open-source/mainframe-ubuntu-linux/9491>

# ANEXOS

**ENTREVISTA**  
**Constructora Proarco S.A.C**

Entrevistado: Consultor de TI de la Constructora Proarco S.A.C

Entrevistadores: Br. Luz Stefany Cacho Aniceto y Br. Joaquín Osmín Aarón Torres Sáenz

¿Cuál es el nombre del proyecto?

---

¿Cuál es la fecha de inicio y término del proyecto?

---

¿Cuál es la ubicación del proyecto?

---

¿Cómo parte la iniciativa de implementar la domótica en Trujillo?

---

¿Qué desean que se mejore para el departamento domotizado?

---

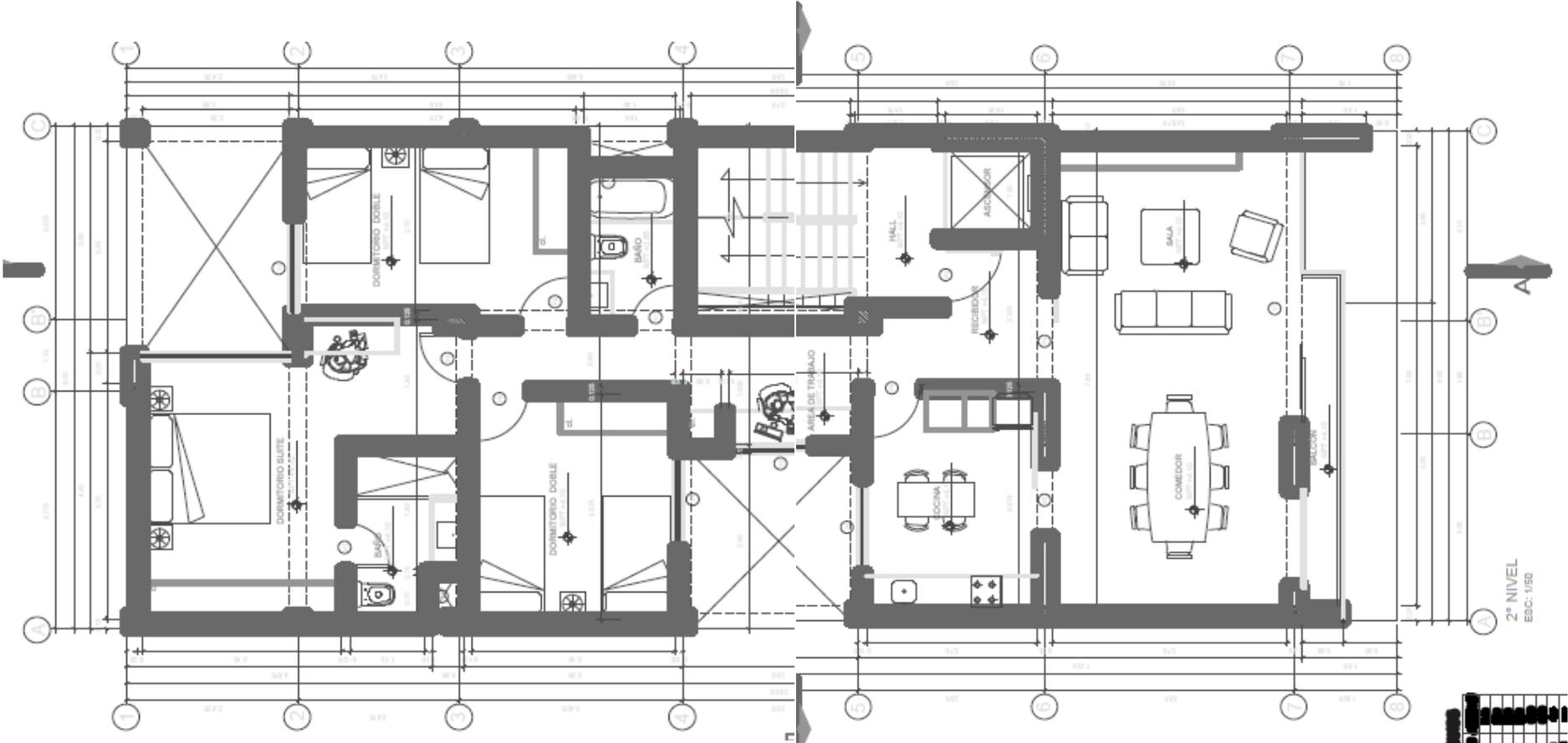
¿Cuál es la cantidad de habitantes en el departamento?

---

¿Cuáles son las características esenciales de la(s) persona(s) que estarán habitando el departamento?

---

ANEXO 02: Planos de un nivel tipo del edificio multifamiliar Jardines del Golf de PROARCO S.A.C.



ANEXO 03: Planos en 3D de un nivel tipo del edificio multifamiliar Jardines del Golf de PROARCO S.A.C.



