

MÁSTER EN INTELIGENCIA ARTIFICIAL

INGENIERÍA ONTOLÓGICA

---

# **Diseño de una ontología para instalaciones deportivas**

---

*Autores*

LUIS COUTO SELLER  
IRENE MARBÁN ÁLVAREZ  
AÍDA MUÑOZ MONJAS

November 6, 2022

# Contenidos

<b>1</b>	<b>Introducción</b>	<b>2</b>
<b>2</b>	<b>Metodología NeOn</b>	<b>2</b>
<b>3</b>	<b>Especificación de la ontología</b>	<b>4</b>
3.1	Propósito . . . . .	4
3.2	Ámbito . . . . .	4
3.3	Lenguaje de la Implementación . . . . .	4
3.4	Potenciales usuarios . . . . .	4
3.5	Potenciales casos de uso . . . . .	5
3.6	Requisitos de la ontología . . . . .	5
3.6.1	Requisitos no funcionales . . . . .	5
3.6.2	Requisitos funcionales: preguntas de competencia . . . . .	5
3.7	Glosario de términos . . . . .	6
<b>4</b>	<b>Planificación temporal de la ontología</b>	<b>7</b>
<b>5</b>	<b>Búsqueda de ontologías de alto nivel</b>	<b>8</b>
<b>6</b>	<b>Recursos no ontológicos</b>	<b>9</b>
<b>7</b>	<b>Patrones en la ontología</b>	<b>10</b>
<b>8</b>	<b>Modelo conceptual</b>	<b>11</b>
8.1	Organización . . . . .	11
8.2	Instalación Deportiva . . . . .	12
8.3	Servicio Ofrecido . . . . .	13
8.4	Regla . . . . .	14
<b>9</b>	<b>Implementación de la ontología con OWL</b>	<b>15</b>
<b>10</b>	<b>Evaluación de la ontología con OOPS!</b>	<b>17</b>
10.1	Mejoras implementadas tras la sugerencia de OOPS! . . . . .	17
10.2	Resultados finales de la evaluación . . . . .	18
<b>11</b>	<b>Documentación de la ontología</b>	<b>18</b>
<b>12</b>	<b>Conclusiones</b>	<b>18</b>
	<b>Bibliografía</b>	<b>19</b>
	<b>Anexos</b>	<b>20</b>

# 1 Introducción

En el contexto de la iniciativa OpenCityData [1] y del proyecto Ciudades Abiertas [2] se han desarrollado un buen número de ontologías que permitan representar datos abiertos de ciudades, en muy diversas áreas (transporte, gestión económica, equipamientos, servicios, etc.). Dentro de esta iniciativa se encuentran muchas áreas todavía por desarrollar, y es una de ellas en las que se enmarca el desarrollo de nuestro trabajo.

El objetivo es diseñar e implementar una ontología que represente de manera adecuada instalaciones deportivas, sus características y acciones relacionadas. Las ontologías y otras fuentes de conocimiento utilizadas durante el desarrollo de este trabajo serán citadas, y se puede acceder a ellas a través de los hipervínculos de la bibliografía. La ontología por desarrollar constituirá un modelo para representar los datos asociados a una instalación deportiva, donde se entiende como instalación deportiva un recinto o construcción provista de los medios necesarios para la práctica, aprendizaje y competición de uno o más deportes, y que está asociada, organizada y mantenida por una determinada organización. Como iremos describiendo en los siguientes apartados de la memoria, se especificarán todos los requisitos necesarios para la creación de la ontología, y se detallarán todas las características de esta.

A la hora de la creación de la ontología se han seguido las pautas que marca la metodología NeOn, haciendo uso tanto de recursos ontológicos como no ontológicos para su reutilización y reingeniería. Cada una de las actividades de la metodología han sido realizadas en una secuencia especificada y con un planteamiento ordenado, y tras ello se ha usado la herramienta para la formalización en OWL de la ontología y OOPS! para su evaluación.

## 2 Metodología NeOn

La Metodología NeOn para la construcción de redes de ontologías es una metodología basada en escenarios que se apoya en los aspectos de colaboración de desarrollo de ontologías y la reutilización, así como en la evolución dinámica de las redes de ontologías en entornos distribuidos [3].

Como define esta metodología, hay un conjunto de nueve escenarios para la construcción de ontologías, haciendo hincapié en la reutilización de recursos ontológicos, la reingeniería y la fusión. Para este trabajo, en el cual se ha seguido la metodología NeOn se han seguido los siguientes escenarios:

- **Escenario 2: La reutilización y reingeniería de los recursos no ontológicos (NOR).** Se llevan a cabo procesos de reutilización NOR para decidir en base a los requisitos definidos de la ontología que NORs se utilizarán para construir la red de la ontología. Después de ello estos NORs se readaptarán mediante re-ingeniería.
- **Escenario 4: La reutilización y re-ingeniería de los recursos ontológicos.** Se llevan a cabo procesos de reutilización de recursos y se reorganizan los recursos ontológicos.
- **Escenario 7: Reutilización de los patrones de diseño de ontologías (ODPs).** Los desarrolladores de ontologías acceden a repositorios de reutilización ODPs.

- **Escenario 9: Localización de recursos ontológicos.** Los desarrolladores de ontologías adaptan una ontología a otras lenguas y la cultura las comunidades, obteniendo así una ontología multilingüe.

Otro punto importante dentro de la metodología NeOn es el glosario de actividades [4]. Durante todo el proceso de construcción de nuestra ontología se han realizado actividades incluidas en este glosario. A continuación se listan algunas de las más importantes:

- **Ontology Requirements Specification.** Planteamiento de preguntas y la resolución de estas para detectar las necesidades y requisitos de la ontología.
- **Ontology Aligning.** Buscamos entre los diferentes recursos ontológicos los elementos comunes para evaluar cual es más adecuada para nuestra ontología.
- **Ontology Annotation.** A los elementos de nuestra ontología les hemos asignado etiquetas (labels) y comentarios para enriquecerla.
- **Ontology Comparison.** A la hora de buscar ontologías ya creadas sobre el mismo dominio tuvimos que comparar y evaluar las diferencias entre ellas para elegir las mas adecuadas.
- **Ontology Conceptualization.** Durante el proceso de adquisición tuvimos que organizar toda la información recopilada para crear un modelo conceptual que cubriese los requisitos definidos.
- **Control.** Se siguió la planificación temporal prevista para completar las tareas de desarrollo programadas.
- **Ontology Design Pattern Reuse.** Se han reusado patrones de diseño para nuestra ontología, concretamente el patrón de Normas y el de Servicios.
- **Ontology Diagnosis.** Gracias al uso de herramientas como OOPS! pudimos evaluar los fallos de la ontología y corregirlos durante el proceso de desarrollo.
- **Ontology Elicitation.** Se realizó una adquisición de estructuras conceptuales como el T-Box de documentos de expertos del dominio de las instalaciones deportivas.
- **Ontology Enrichment.** Hemos extendido con más relaciones y clases algunas de las ontologías que hemos reusado para completarlas para cubrir ciertos requisitos.
- **Ontology Implementation.** Una vez diseñado el modelo conceptual se transformó la ontología a un modelo formal (código OWL) gracias a Protégé.
- **Ontology Localization.** Las entidades y relaciones creadas se etiquetaron tanto en español como inglés.
- **Ontology Merging.** Se fusionaron clases y funciones de distintas ontologías para crear una nueva.
- **Ontology Search.** Se realizó una búsqueda de ontologías o módulos de ontologías candidatas para ser reutilizadas.

- **Ontology Module Extraction.** Se han extraído módulos concretos de ciertas ontologías.
- **Ontology Module Reuse.** Se han reutilizado los módulos extraídos de otras ontologías.
- **Non-Ontological Resource Reengineering.** Se han reestructurado contenidos no ontológicos para adaptarlos a nuestra ontología.
- **Non-Ontological Resource Reuse.** Se han usado recursos no ontológicos para transformarlos en partes de ontologías.
- **Scheduling.** Se ha realizado una planificación de las actividades anteriores para completarlas en el orden correcto y así llevar un control de las mismas.

### 3 Especificación de la ontología

#### 3.1 Propósito

El objetivo de esta Ontología se enmarca en el proyecto Ciudades abiertas, que tiene como objetivo representar datos de ciudades en diversas areas. el objetivo de esta ontología sera modelar el conocimiento de Instalaciones Deportivas para que pueda ser utilizado como ontología dentro del proyecto.

#### 3.2 Ámbito

La definición de Instalaciones Deportivas puede llegar a ser ambigua, dado que existen multiples instalaciones donde se realizan deporte con un fin muy distinto: un estadio de futbol y un parque de barras podrían ser ambos considerados instalaciones deportivas.

Nuestra ontología se enmarca en la definición propuesta por el informe estadístico de 2022 publicado por el Ministerio de Educación, Cultura y Deporte, donde se proporcionan indicadores estadísticos para estimar las dimensiones y características de las infraestructuras en España. En el documento, se define instalación deportiva como: 'Instalaciones destinadas al deporte que incluyen uno o varios espacios deportivos donde puede desarrollarse la actividad físico-deportiva.'.

#### 3.3 Lenguaje de la Implementación

La ontología sera implementada en OWL utilizando .

#### 3.4 Potenciales usuarios

Los usuarios identificados como posibles interesados en el uso de la ontología han sido los siguientes.

1. Ayuntamiento interesado en el proyecto ciudades abiertas que quiera aumentar la transparencia de los datos de sus instalaciones deportivas.
2. Persona responsable de una instalación deportiva que quiera almacenar estadísticas de la misma y/o crear una página web de la instalación.

3. Deportista que quiera conocer que instalación deportiva le es más conveniente.
4. Ministerio u órgano del gobierno que quiera analizar estadísticas sobre el uso de las instalaciones deportivas en su país.

### 3.5 Potenciales casos de uso

Los posibles casos de uso identificados son los siguientes.

1. Conocer los subespacios forman una Instalación Deportiva concreta.
2. Buscar servicios deportivos como clases o alquileres de espacios ofrecidos por la organización que gestiona la Instalación Deportiva.
3. Enumerar los deportes que pueden realizarse en cada Instalación Deportiva, y en que subespacio se realizan.
4. Obtener estadísticas del uso de servicios ofrecidos en la Instalación Deportiva por cliente.
5. Obtener información sobre los trabajadores de la organización que lleva la instalación deportiva.

### 3.6 Requisitos de la ontología

#### 3.6.1 Requisitos no funcionales

Los requisitos no funcionales de la ontología se refieren a las características y aspectos generales no directamente relacionados con el contenido de la ontología. En nuestro caso son los siguientes:

- NFR1: La ontología deberá soportar otros idiomas a parte de Español.
- NFR2: La documentación asociada a la ontología debe incluir fuentes fiables.
- NFR3: El formato utilizado para representar el conocimiento será OWL.
- NFR4: El sistema de nombrado en la ontología sera comprensible y coherente.

#### 3.6.2 Requisitos funcionales: preguntas de competencia

Para las preguntas de competencia, hemos optado por un acercamiento *Middle out*. Comenzamos realizando preguntas *Bottom-Up*, sin mirar bases de datos ni contenido relacionado, simplemente buscando preguntas sencillas que podrían estar relacionadas con una instalación deportiva. Cuando llegamos a un limite donde nuestras preguntas carecían de complejidad, entonces tuvimos un acercamiento *Top-Down*, donde ya comenzamos a mirar información acerca de Instalaciones Deportivas, y pudimos desarrollar preguntas más complejas. El resultado puede verse en la tabla:

<b>CQ1</b>	¿Cómo se llama la instalación deportiva?
<b>CQ2</b>	¿Cuál es la geolocalización de la instalación deportiva?
<b>CQ3</b>	¿Cuál es la dirección de la instalación deportiva?
<b>CQ4</b>	¿Cuál es el horario de apertura de la instalación deportiva?
<b>CQ5</b>	¿Cómo se llama la organización que gestiona la instalación deportiva?
<b>CQ6</b>	¿Quién es el director de la organización que gestiona la instalación deportiva?
<b>CQ7</b>	¿Cuántos entrenadores se dedican al baloncesto?
<b>CQ8</b>	¿Qué tipo de trabajadores tiene la organización que gestiona la instalación deportiva?
<b>CQ9</b>	¿Cuál es el número de teléfono para contactar con la organización de la instalación deportiva?
<b>CQ10</b>	¿Qué roles puede tener un entrenador?
<b>CQ11</b>	¿La instalación deportiva tiene baños o vestuarios?
<b>CQ12</b>	¿Qué espacios deportivos tiene al aire libre?
<b>CQ13</b>	¿Qué espacios deportivos cubiertos tiene?
<b>CQ14</b>	¿La instalación deportiva tiene piscina?
<b>CQ15</b>	¿Cuántas pistas de fútbol tiene la instalación deportiva?
<b>CQ16</b>	¿Qué deportes se pueden realizar en la instalación deportiva?
<b>CQ17</b>	¿La instalación deportiva tiene cafetería?
<b>CQ18</b>	¿Qué servicios de día (individuales) se pueden comprar en la instalación deportiva?
<b>CQ19</b>	¿Qué monitores imparten las clases de zumba que ocurren en la sala multiusos?
<b>CQ20</b>	¿Cuánto cuesta alquilar una pista de tenis durante 1 hora?
<b>CQ21</b>	¿Qué días hay clases de tenis?
<b>CQ22</b>	¿A qué hora son las clases de natación los lunes?
<b>CQ23</b>	¿Qué bonos de servicios son ofrecidos por la organización de la instalación deportiva?
<b>CQ24</b>	¿Cuánto cuesta un bono mensual de sala de musculación?
<b>CQ25</b>	¿Cuánto dura la clase de baloncesto impartida por un monitor concreto?
<b>CQ26</b>	¿Qué reglas de vestimenta tiene la piscina de la instalación deportiva?
<b>CQ27</b>	¿A quién aplica la regla de vestimenta de la piscina?
<b>CQ28</b>	¿Cuántos clientes han contratado el bono de clases de zumba el mes de enero?

Figure 1: Preguntas de competencia

### 3.7 Glosario de términos

Tras realizar las preguntas de competencia, hemos contado los sustantivos relevantes que se repetían, tanto en las preguntas para identificar clases y relaciones, como en las respuestas.

Cabe destacar que las preguntas se han contestado de manera muy genérica, sin concretar para una Instalación específica, de forma que los términos recogidos están un nivel por encima de ser instancias. Los resultados de algunos de los términos más relevantes se recogen en la siguiente figura:

Pre glossary of terms			
Terms from questions + Frequency			
Instalación deportiva	16	Dirección/geolocalización	2
Deportes	8	Trabajadores	4
Organización	5	Regla	1
Clases	4	Pistas	1
Bono	4	Baño/Vestuario	2
Horario/hora/día	6	Piscina	1
Servicios	2	Aire libre / Cubierto	1
Cientes	1	Alquiler	1
Cuesta (precio)	1	Cafetería	1
Terms from answers + Frequency			
Nombre Instalación	1	Cantidad de dinero	1
Coordenadas	1	Entrenador/monitor/técnico ...	1
Teléfono	1	Zumba/Baloncesto/Tenis	1
Localidad/Código Postal...		Franja horaria	1
Pista de Baloncesto, Espacio Multiusos, Sala musculación	1	Bono de clases	1

Figure 2: Glosario de términos

## 4 Planificación temporal de la ontología

El ciclo de vida utilizado en el diseño y desarrollo de esta ontología es el ciclo de vida incremental, pudiéndose considerarse como la primera iteración de un modelo ágil.

Se decidió utilizar un ciclo de vida incremental debido a las características del proyecto, ya que la ontología no fue diseñada a partir de un set de datos proporcionados por el cliente, si no que se realizó el modelado a partir de las posibles necesidades de sus usuarios.

La planificación temporal se puede dividir en varias secciones. Durante la planificación de la ontología se buscó el ámbito sobre el que realizar el trabajo, un tema sobre el que no existiesen ontologías ya diseñadas, y se definieron los requisitos planteando las preguntas de competencia. Se diseñó la ontología reutilizando los recursos útiles encontrados plasmando en el modelo conceptual. A continuación se creó la implementación del modelo y su evaluación. La redacción de este documento se puede considerar un ejercicio transversal.



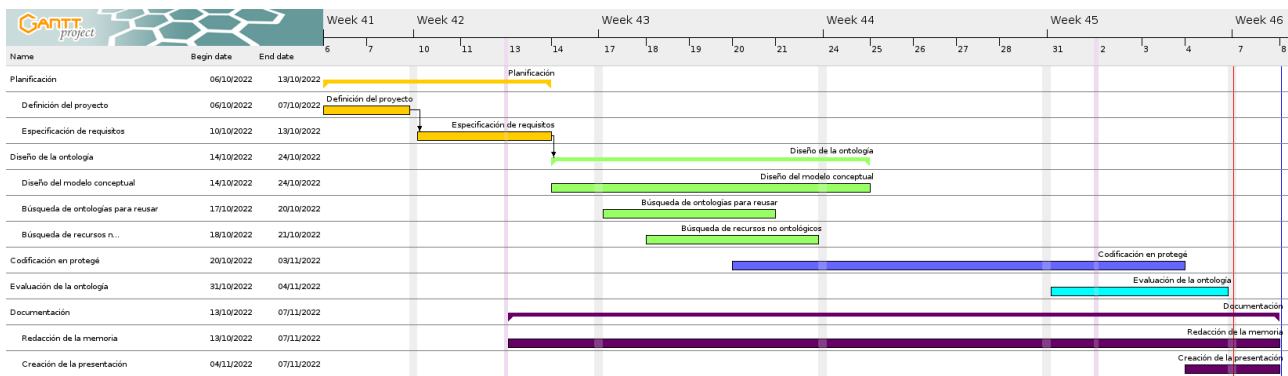


Figure 3: Diagrama de Gantt de la planificación temporal seguida.

## 5 Búsqueda de ontologías de alto nivel

Una vez identificados el glosario de términos mediante la especificación, y creado un modelo donde se relacionan las diferentes entidades, el siguiente paso es la formalización de la ontología. Para ello, hemos realizado una búsqueda de ontologías de alto nivel, detectando aquellas que cubren de manera total o parcial secciones de nuestro modelo para poder reutilizarlas.

Las primeras ontologías que hemos identificado para nuestra ontología, fueron mencionadas a lo largo del curso:

- **Org.** La ontología org ha sido utilizada para representar la entidad de la organización que lleva la instalación deportiva. No hemos utilizado la ontología completa, solo los módulos que nos hacían falta para representar la organización, su localización y sus miembros. Esta ontología es ampliamente utilizada: en LOV, podemos ver que 37 ontologías reutilizan contenido de org.
- **Foaf.** Para poder representar a todas las personas que pertenecen a nuestro modelo (director, entrenadores, monitores, deportistas...) hemos utilizado foaf. A efectos prácticos, nos ha servido con importar la ontología org, ya que esta contenía las entidades foaf que nos hacían falta
- **Odrl.** La ontología ODRL permite modelar reglas y políticas. Se ha utilizado para representar las reglas de vestimenta en los espacios deportivos. Para ello, se ha utilizado únicamente parte de la ontología, en concreto las entidades que son necesarias para generar el patrón Norm, que describiremos más adelante.
- **Geo.** La ontología Geo se ha utilizado para poder expresar la geolocalización de la instalación mediante la entidad GeoPoint. Esta ontología no tiene propiedades, así que hemos tenido que crear una propiedad que relacione la instalación con el GeoPoint.
- **Vcard.** Consiste en una ontología utilizada para representar la información que contiene una tarjeta de visita. En nuestro caso, hemos utilizado únicamente las entidades referentes a la dirección (Address) y nombre (OrganizationName).

Una vez identificadas estas ontologías de alto nivel, utilizamos el buscador de LOV (Linked Open Vocabularies) para buscar otras posibles ontologías que pudiéramos reutilizar. En concreto encontramos las siguientes:

- A la hora de importar todas estas ontologías en `LOV`, utilizamos `LOV` como método de búsqueda de las URIs, lo que facilitó la tarea.

Una de las fuentes de información utilizadas para generar la ontología propuesta en este documento es el siguiente informe [5]. A partir de este recurso no ontológico, mediante el uso de un T-Box se pudo realizar una modelización de la información descrita que representa la estructura taxonómica del documento en esta ontología de instalaciones deportivas.

Al plasmar el conocimiento presente en este documento en el modelo de la ontología mediante un T-Box, se decidió mantener las clases intermedias presentes en el documento como parte de la ontología pese a que serán probablemente rara vez utilizadas para mantener la estructura del documento mencionado y facilitar la reutilización de este.



9

## 7 Patrones en la ontología

A la hora de diseñar nuestra ontología, de entre los patrones de diseño ya existentes vimos que algunos de ellos eran aplicables a nuestro caso de instalaciones deportivas. Concretamente, los patrones que hemos implementado han sido el de “Servicios ofrecidos” y el de “Normas”.

Para el diseño de los servicios ofrecidos por la organización en una instalación deportiva, se reutilizó el patrón de una relación N-aria según lo visto en clase. Este patrón se utiliza para representar una relación N-aria en el que todos los elementos tienen la misma importancia. Para representar esta relación N-aria se crea una clase, en nuestra ontología la clase *ServicioOfrecido*, a la que asocia todos los atributos de la relación. Este patrón se utilizará para representar los servicios que una organización ofrece que se realizarán en una instalación deportiva en un horario determinado y por un precio indicado.

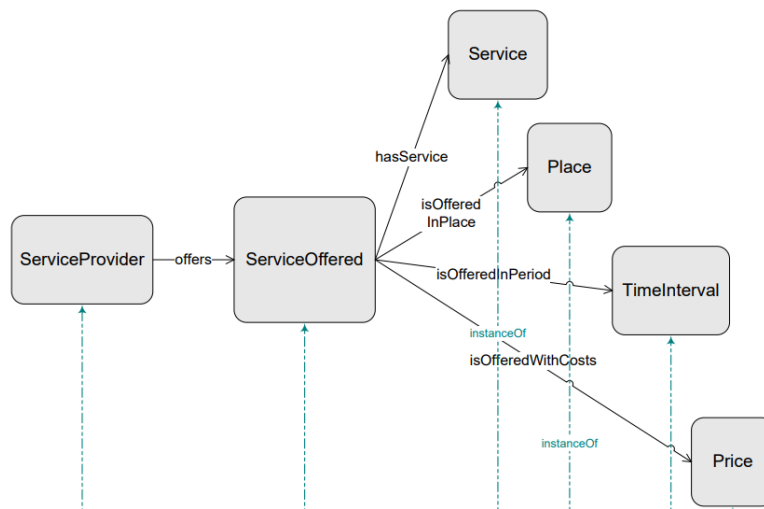


Figure 5: Esquema del patrón de servicios.

El patrón de normas está destinado a la definición de una serie de reglas y normas que aplican a la organización. El patrón de reglas se expresa mediante ODRL (Open Digital Rights Language), y se definen mediante la estructura: “Una *acción* es *permitida/obligatoria/prohibida* de ser realizada por una *parte* sobre el *activo*, siempre que se mantengan las *restricciones*”.

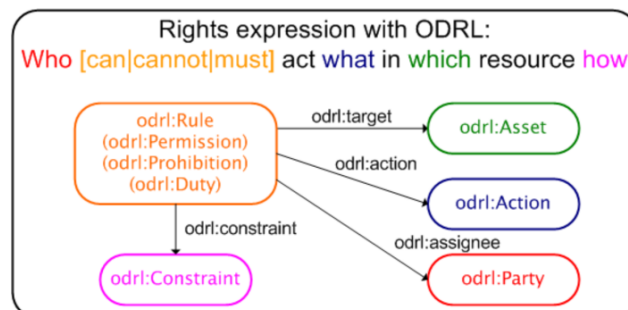


Figure 6: Esquema del patrón de servicios.

En la sección de modelado conceptual de la ontología se entrará más en detalle sobre las clases y relaciones de estos patrones.

## 8 Modelo conceptual

Tras el proceso de búsqueda de recursos ontológicos y no ontológicos para su reutilización como se ha explicado en los apartados anteriores así como la selección de patrones de diseño de ontologías, se desarrolló el modelo conceptual de la ontología. Esta es una de las fases más importantes en la construcción de ontologías, y en este apartado se describirán las clases y relaciones que se han implementado y que en los siguientes apartados se formalizarán a código OWL. Debido a la amplitud del modelo y para un mejor entendimiento se procederá a explicar los distintos módulos del modelo por separado.

Encontramos cuatro clases principales que serán el núcleo de nuestra ontología. Estas clases son **Organización**, **Instalación Deportiva**, **Servicio Ofrecido** y **Regla**.

El diagrama completo de la ontología está disponible en el anexo.

### 8.1 Organización

En lo que a la organización respecta, reutilizando clases y relaciones de la ontología **org** hemos creado el modelo que se puede observar en la figura a continuación. La forma de interpretar esta parte del modelo es: “Una organización encargada de gestionar instalaciones deportivas establece acuerdos de membresía con sus trabajadores para que estos tengan un determinado rol dentro de la organización”. Analizando el modelo con sus clases y relaciones tenemos:

- Una organización es *SubClassOf* de *org:Organization*. Organización tiene como propiedades el Nombre, Email y Número de teléfono, que son literales.
- Una organización es la que gestiona una o más instalaciones deportivas, por lo que la clase Instalación Deportiva se relaciona con Organización mediante la relación *managedBy*.
- Una organización establece unos acuerdos de membresía con sus trabajadores para un determinado rol (tal y como está definido en la ontología *org*) por lo que tenemos la clase *org:Membership* que se relaciona con organización mediante la relación *org:organization*, se relaciona con las clases de los trabajadores mediante la relación *org:memberOf* y se relaciona con *org:Role* mediante la relación *org:role*. Al diseñar esta sección hemos querido contemplar la posibilidad de que un trabajador de la organización tenga asignada la instalación o las instalaciones en las que trabaja por lo que hemos unido *org:Membership* con Instalación Deportiva mediante la relación *establecidaEn*.
- Para determinar los tipos de trabajadores que hay en una organización hemos decidido crear una clase para cada uno de los tipos. Debido a la gran cantidad de clases de trabajadores que existen para su simplificación en nuestro modelo hemos ejemplificado las clases Monitor, Limpiador, y Entrenador, todas ellas *SubClassOf* de *foaf:Person*.

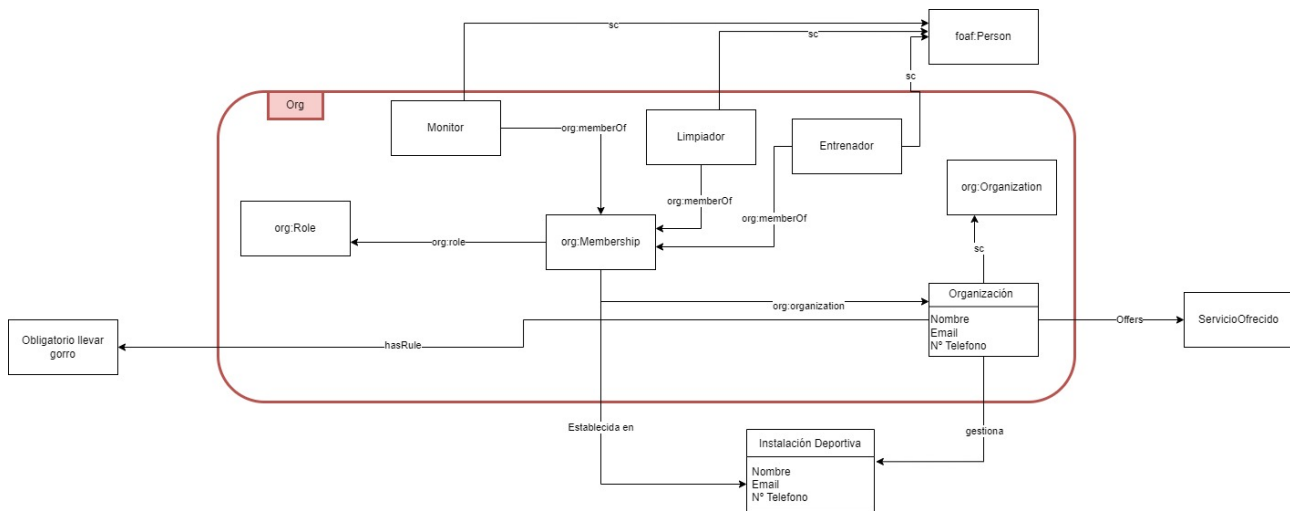


Figure 7: Esquema de las principales clases relacionadas con **Organización**.

## 8.2 Instalación Deportiva

Para la construcción de la taxonomía de las instalaciones deportivas se ha hecho uso del recurso no ontológico explicado en apartados anteriores, y mediante un T-Box se ha modelado a nuestra ontología.

- Una Instalación Deportiva puede estar compuesta por diferentes partes (*partOf*): Espacio Deportivo (que puede ser al aire libre o cerrado), Espacio Complementario, Espacio Auxiliar. Estas partes son disjuntas unas de otras. Un Espacio Complementario va asociado a un Espacio Deportivo.
- Cada una de estas partes tiene su propia taxonomía tal y como se define en el recurso no ontológico y se muestra las diferentes subclases y las clases que son disjuntas (recuadros rosas) en el esquema de la Figura X2.
- La clase Instalación Deportiva cuenta con propiedades como número de teléfono e email que serán literales y no clases.
- Una instalación deportiva se geolocaliza en un *geo:Point* mediante la relación *hasGeopoint*. A su vez expresamos su localización mediante una dirección, para ello relacionamos Instalación Deportiva con la clase *vcard:Address* mediante la relación *vcard:hasAddress*.
- La clase Deporte se relaciona con un Espacio Deportivo mediante la relación *esPracticadoEn*.
- La clase Cliente es *subClassOf* *foaf:Person*, y Cliente practica uno o varios Deportes. En el siguiente modulo también se verá que Cliente consume ServicioOfrecido
- Entrenador y Monitor (definidos en el apartado de organización) tienen una relación con Deporte que es *enseña*.

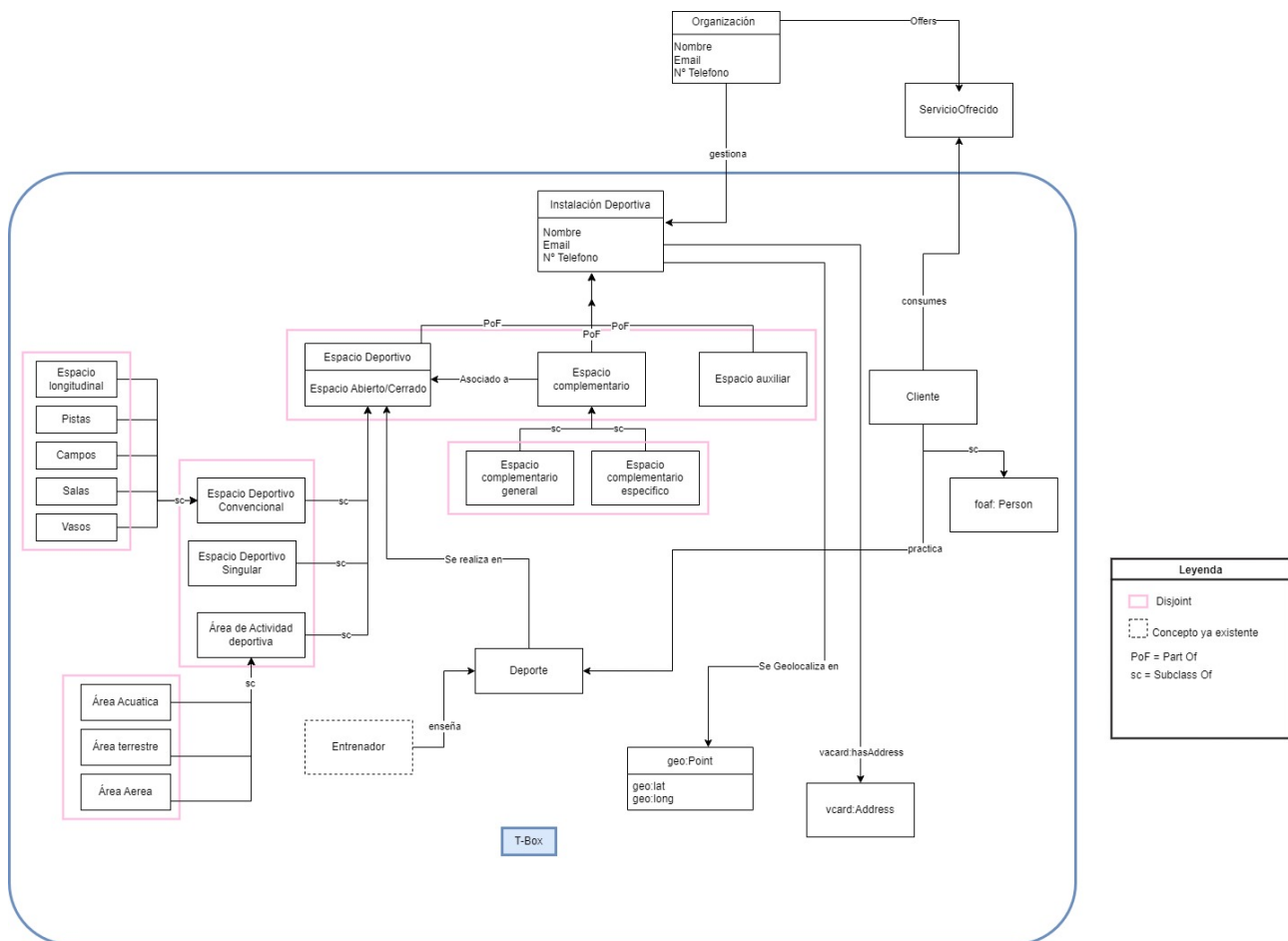


Figure 8: Esquema de las principales clases relacionadas con **Instalación Deportiva**.

### 8.3 Servicio Ofrecido

Esta clase ServicioOfrecido tendrá asociadas varias clases mediante diferente relaciones:

- **Servicio**, que tiene dos subclases disjuntas Individual y Bono de servicios. De estas dos clases habría más subclases, para ejemplificar hemos incluido el alquiler de una pista de pádel o un bono de 10 clases de zumba
- **Instalación Deportiva** en la que se ofrece dicho servicio. Por ejemplo, una pista de pádel, o la sala de musculación.
- **time:DurationDescription** y **time:GeneralDateTimeDescription**, que indicaran la duración y/o la fecha en la que se ofrece el servicio. Por ejemplo, jueves 3 de noviembre del 2022 de 18:00 a 19:00, o Todos los lunes y miércoles de febrero de 15:00 a 16:00.
- **PriceSpecification**, teniendo asociada como propiedades una cantidad y una moneda de pago. Por ejemplo 59,99 euros.

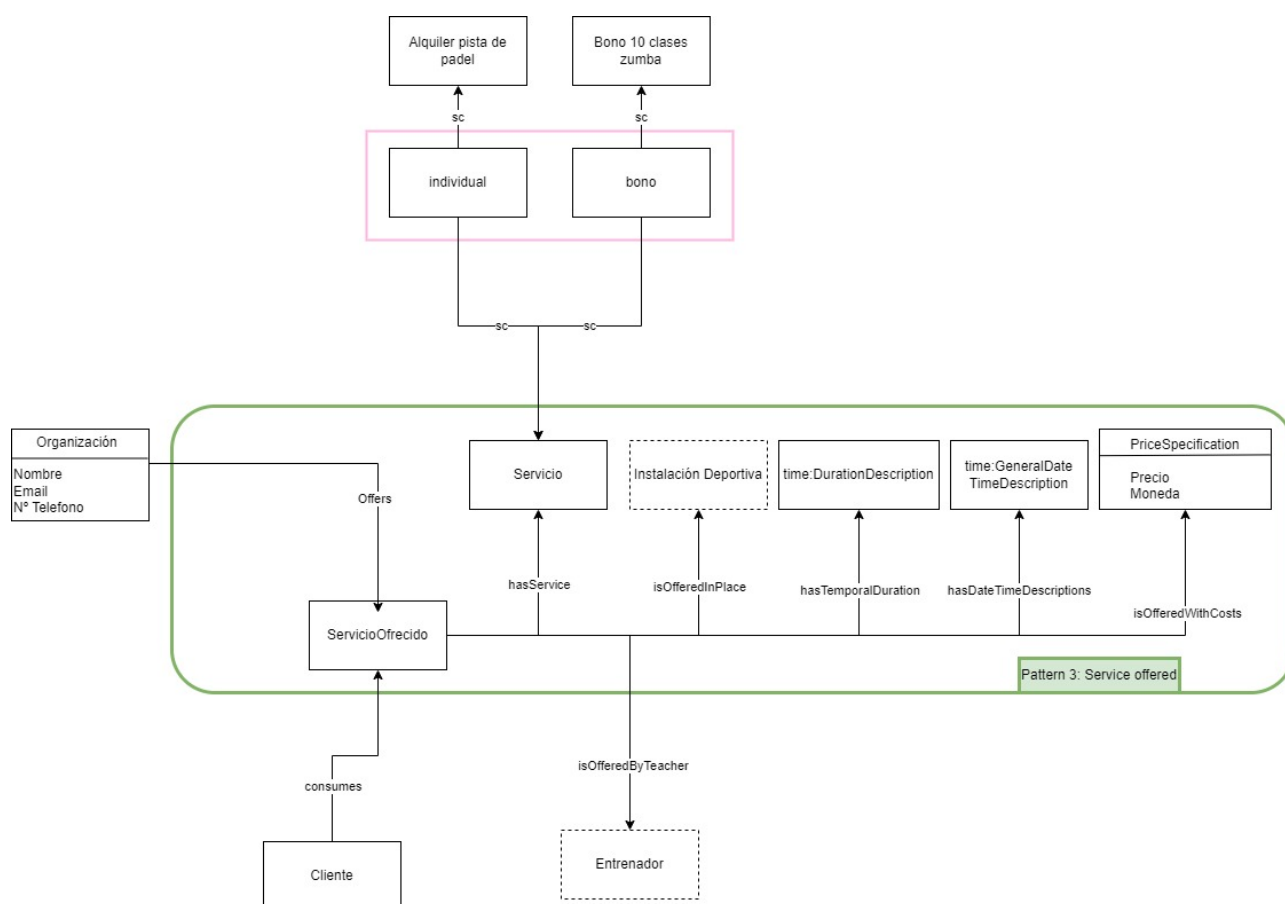


Figure 9: Esquema de las principales clases relacionadas con **Servicio Ofrecido**.

## 8.4 Regla

Al incluir este patrón de normas, hemos creado una única regla para ejemplificar las normas de una organización (en la realidad habría muchas de ellas, pero para simplificar el modelo para este trabajo se ha trabajado solo con una). Las reglas constan de la clase Rule que define si se trata de una prohibición, una obligación o permiso, y de ella salen varias relaciones a una serie de clases que serán subclass of de las siguientes entidades de la ontología ODRL:

- **odrl:Constraint.** Para cada norma hay una subclass de *odrl:Constraint* y puede ser un lugar físico (piscina, pista de tenis, gimnasio) o un marco temporal (los sábados, el día 1 de cada mes, todos los días de 9:00 a 10:00) en el que se aplica la norma.
- **odrl:Asset.** Para cada norma tendremos una clase *subclass of odrl:Asset* que establece el activo o recurso que es sujeto de la norma. Este activo puede ser cualquier forma de recurso identificable, como datos/información, servicios o elementos físicos. En nuestra ontología hemos puesto el ejemplo de gorro de baño.
- **odrl:Action.** Para cada norma tendremos una clase *subclass of odrl:Action* que indicará la acción realizada sobre el activo. En nuestro modelo encontramos la clase Llevar puesto para el caso de gorro de baño.

- **odrl:Party** Tendremos una *subclass* de esta clase de ODRL para cada regla, y definirá a que parte de las personas se le aplica dicha regla. Puede ser por ejemplo una regla que aplique a todos los usuarios, o una regla que aplique solo a trabajadores.

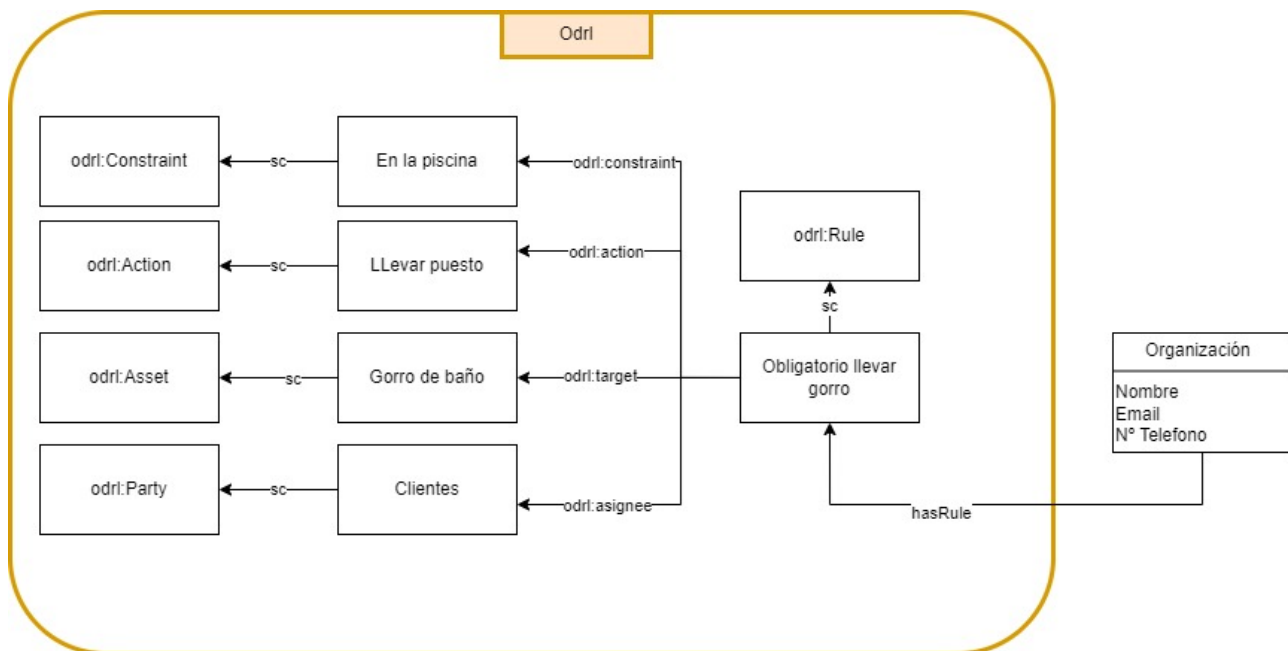


Figure 10: Esquema de las principales clases relacionadas con **Regla**.

## 9 Implementación de la ontología con OWL

Una vez identificados el glosario de términos mediante la especificación, y hemos conceptualizado un modelo donde se relacionan las diferentes entidades, el siguiente paso, la formalización de la ontología. Para ello, hemos realizado una búsqueda de ontologías de alto nivel, de forma que si alguna de ellas cubre toda o parte de alguna de nuestras entidades, podamos **reutilizarla**.

Las primeras ontologías que hemos identificado para nuestra ontología, fueron mencionadas a lo largo del curso:

- **Org:** La ontología org ha sido utilizada para representar la entidad de la organización que lleva la instalación deportiva. No hemos utilizado la ontología completa, solo los módulos que nos hacían falta para representar la organización, su localización y sus miembros. Esta ontología es ampliamente utilizada: en LOV, podemos ver que 37 ontologías reutilizan contenido de org.
- **Foaf:** Para poder representar a todas las personas que pertenecen a nuestro modelo (director, entrenadores, monitores, deportistas...) hemos utilizado foaf. A efectos prácticos, nos ha servido con importar la ontología org, ya que esta contenía las entidades foaf que nos hacían falta
- **Odrl:** La ontología ODRL permite modelar reglas y políticas. Se ha utilizado para representar las reglas de vestimenta en los espacios deportivos. Para ello, se ha utilizado



únicamente parte de la ontología, en concreto las entidades que son necesarias para generar el patrón Norm, que describiremos más adelante.

- **Geo:** La ontología Geo se ha utilizado para poder expresar la geolocalización de la instalación mediante la entidad GeoPoint. Esta ontología no tiene propiedades, así que hemos tenido que crear una propiedad que relacione la instalación con el GeoPoint.
- **Vcard:** Consiste en una ontología utilizada para representar la información que contiene una tarjeta de visita. En nuestro caso, hemos utilizado únicamente las entidades referentes a la dirección (Address) y nombre (OrganizationName).

Una vez identificadas estas ontologías de alto nivel, utilizamos el buscador de LOV (Linked Open Vocabularies) para buscar otras posibles ontologías que pudieramos reutilizar. En concreto encontramos las siguientes:

- **Time:** La ontología time permite expresar tanto fechas como intervalos y duración de eventos. En nuestro caso, para expresar la fecha y la duración de los servicios hemos utilizado dos entidades diferentes.
- **Sport:** Esta ontología estaba disponible en el buscador, pero las URIs no llevaban a ninguna ontología, por lo que no pudimos reutilizarla.

A la hora de importar todas estas ontologías en , utilizamos LOV como método de búsqueda de las URIs, lo que facilitó la tarea. Como se ha especificado en los requisitos, la ontología estará implementada en OWL. Para eso, hemos utilizado Protégé, un editor de ontologías open source.

El primer paso que hemos hecho ha sido incluir todas las clases de la ontología que hemos creado mediante el T-Box. Después, hemos reutilizado de las ontologías nombradas en el apartado 5 todas las clases que nos hacían falta. Para ello, hemos importado las ontologías con su URI y movido los axiomas necesarios a nuestra ontología.

En general, la metodología para importar las ontologías ha sido idéntica para todas: primero hemos importando la ontología al completo con su uri, moviendo los axiomas que necesitábamos, y después hemos eliminando el resto. La única excepción ha sido la ontología Time.

Hemos importado la ontología Time de forma completa, ya que utilizábamos la totalidad de sus entidades. Al importar la ontología, nos hemos dado cuenta de que muchas de las propiedades estaban incompletas: les faltaba el rango o el dominio. Hemos optado por mantenerla como estaba, ya que muchos de estos problemas se debían a términos deprecados, y no hemos encontrado una versión más reciente de la ontología.

Cabe destacar que todas las clases tienen etiquetas estilo *label* en español e inglés.

Una vez importadas las clases necesarias, hemos definido aquellas que nos hacían falta para completar nuestro modelo, que no se encontraban ni en el T-Box ni en las ontologías de alto nivel, como puede ser por ejemplo la clase Monitor, Entrenador, Pista de Baloncesto...

Este mismo procedimiento lo hemos realizado para los Object Properties. En este caso, a demás de importar los necesarios de las diferentes ontologías y crear los que necesitábamos, hemos definido las propiedades de cada propiedad.

Hemos tenido cierta dificultad a la hora de crear la propiedad Part Of, ya que tiene varios dominios para un único rango (tanto los espacios deportivos, como los complementarios y auxiliares son part of de una instalación deportiva). Para evitar interferencias, hemos incluido los diferentes dominios con la condición OR.

De forma similar hemos añadido los Data Properties. Estos en lugar de apuntar a una entidad, apuntan a un literal, por lo que hemos tenido que especificar los tipos de datos para literales como la divisa, el precio... Un caso particular ha sido la propiedad hasCeiling (tiene techo), para definir los Espacios Deportivos. En este caso el literal era un booleano.

Finalmente, hemos completado las labels y comentarios de aquellas entidades que no han sido directamente importadas, asegurandonos de que estaban disponibles tanto en inglés como español.

En el anexo se puede encontrar una imagen de la jerarquía de las clases, los Object Properties y los Data Properties presentes en la ontología.

Una vez finalizada la implementación de la ontología, la hemos exportado como RDF para poder evaluarla con OOPS!

## 10 Evaluación de la ontología con OOPS!

La ontología diseñada se ha evaluado utilizando la herramienta Ontology Pitfall Scanner! (OOPS!)[6], que identifica errores en la ontología dividiéndolos en críticos, importantes y menores, de acuerdo a una batería errores comunes.

### 10.1 Mejoras implementadas tras la sugerencia de OOPS!

Tras la primera iteración de la evaluación con OOPS!, la herramienta destacó los siguientes problemas críticos:

- **P19:** Propiedades con múltiples rangos o dominios.

Una de las relaciones definidas tenía más de un dominio, habiendo escrito en accidentalmente *and* en lugar de *or* en el dominio.

Los dominios de las relaciones *time:hasDateTimeDescription* y *time:hasTemporalDuration* fueron modificados añadiendo una entrada en sus dominios y eliminando la entrada de la ontología *time*. Esta modificación no fue aplicada por Protégé, y hubo que modificar la entrada del dominio impuesta por importar la ontología *time* para que el cambio quedase registrado.

- **P29:** Relaciones transitivas mal definidas.

Las relaciones *partOf*, *hasService*, *managed\_by* y *constraint* estaban definidas como transitivas, teniendo rangos y dominios distintos. Por definición una propiedad transitiva debe tener el mismo rango y dominio.

En el caso de *partOf*, una relación que generalmente es transitiva, decidimos eliminar esta propiedad, ya que para nuestro modelo aparece únicamente entre Instalación Deportiva y los espacios deportivos, complementarios y auxiliares, que nunca serán una parte de otros.

Las otras relaciones se encontraban en esta clasificación por un error en la transcripción a Protégé.

Las propiedades *vcard:hasEmail* y *vcard:hasTelephone* aparecen como propiedades sin rango (P11), marcado como error importante, por ser su rango una clase deprecada de la ontología *vcard*. Pese a estar deprecada, esto no es un error, por lo que no se solucionó el problema.

Gracias a los errores destacados por OOPS! se pudieron solucionar todos los errores críticos y varios errores clasificados como importantes. Se decidió no solucionar los errores no críticos de las ontologías importadas debido a las restricciones temporales sobre la entrega del trabajo y a la prioridad que se impuso sobre la reutilización de esta ontología.

## 10.2 Resultados finales de la evaluación

Tras aplicar las correcciones sobre los errores críticos indicados por OOPS! la última versión de la ontología tiene 11 errores importantes, 63 errores menores y 16 sugerencias, mientras que en su primera iteración tenía 8 errores críticos, 12 errores importantes y 70 errores menores, así como 16 sugerencias.

Cabe destacar que aproximadamente el 40% de los errores indicados originalmente por OOPS! estaban relacionados con la ontología importada *time*.

Los resultados completos de ambas evaluaciones se pueden encontrar en el Anexo.

## 11 Documentación de la ontología

[7]

## 12 Conclusiones

# Bibliografía

## References

- [1] "OpenCityData," <https://github.com/opencitydata/>.
- [2] "Ciudades abiertas," <https://ciudadesabiertas.es/>.
- [3] Ontology Engineering Group. UPM, "La Metodología NeOn," <https://oeg.fi.upm.es/index.php/es/methodologies/59-neon-methodology/index.html#:~:text=La%20Metodolog%C3%ADa%20NeOn%20para%20la,de%20ontolog%C3%ADas%20en%20entornos%20distribuidos>.
- [4] —, "The terms (and their definitions) included in the neon glossary are," <https://oeg.fi.upm.es/files/pdf/NeOnGlossary.pdf>.
- [5] Ministerio de Cultura y Deporte, "Anuario de Estadísticas Deportivas 2022. Instalaciones y espacios deportivos." <https://www.culturaydeporte.gob.es/dam/jcr:90d8389b-60eb-41b4-96c1-6f4734d356e5/nota-resumen-instalaciones-y-espacios-deportivos.pdf>, 2022, (Accessed on 18/10/2022).
- [6] M. Poveda-Villalón, A. Gómez-Pérez, and M. C. Suárez-Figueroa, "OOPS! (OntOlogy Pitfall Scanner!): An On-line Tool for Ontology Evaluation," *International Journal on Semantic Web and Information Systems (IJSWIS)*, vol. 10, no. 2, pp. 7–34, 2014.
- [7] Garijo, Daniel, "WIDOCO: a wizard for documenting ontologies," in *International Semantic Web Conference*. Springer, Cham, 2017, pp. 94–102. [Online]. Available: <http://dgarijo.com/papers/widoco-iswc2017.pdf>

## Anexos

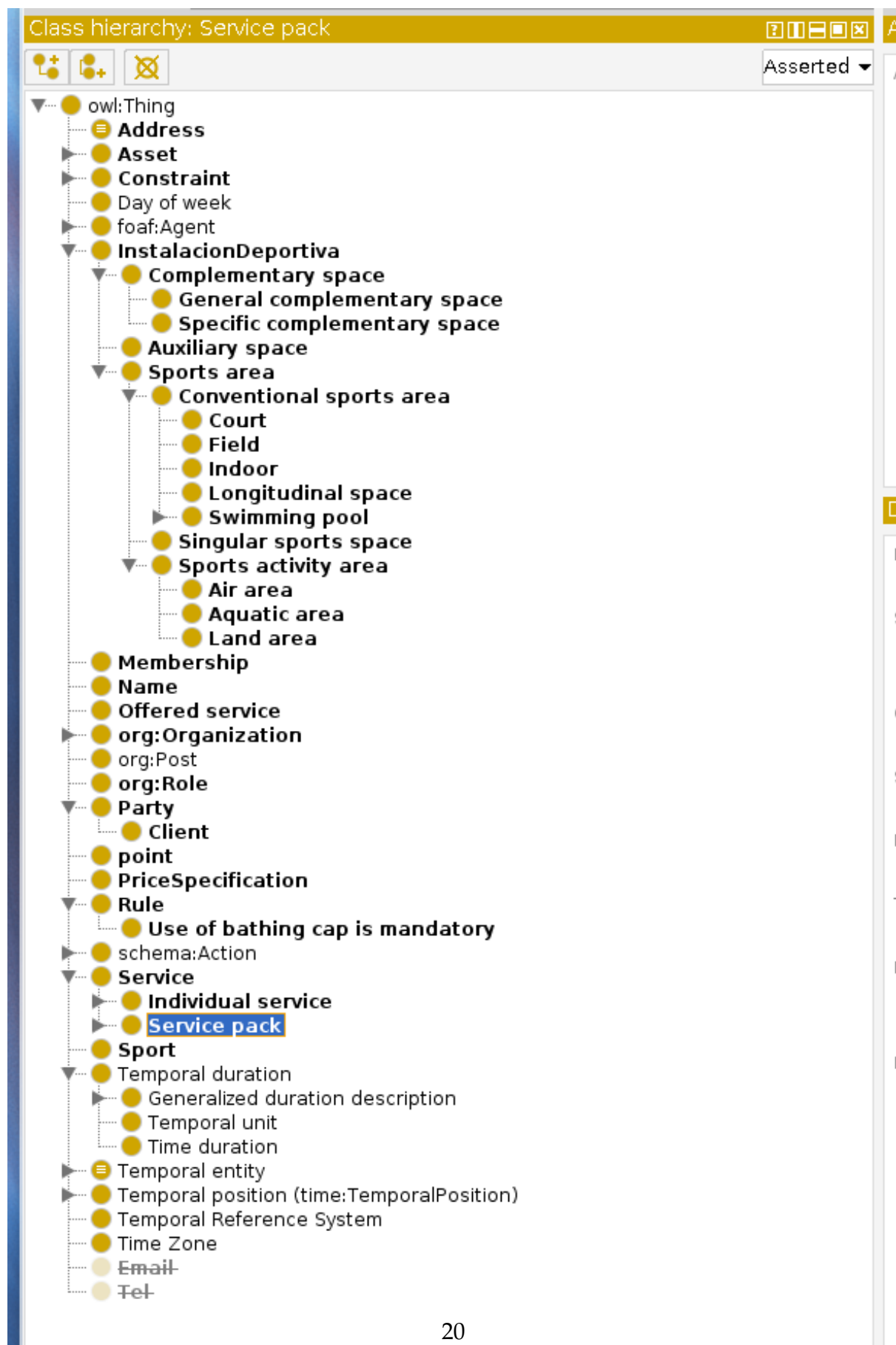


Figure 11: Jerarquía de clases de la ontología.



Figure 12: Object Properties presentes en la ontología.



Figure 13: Data Properties presentes en la ontología.

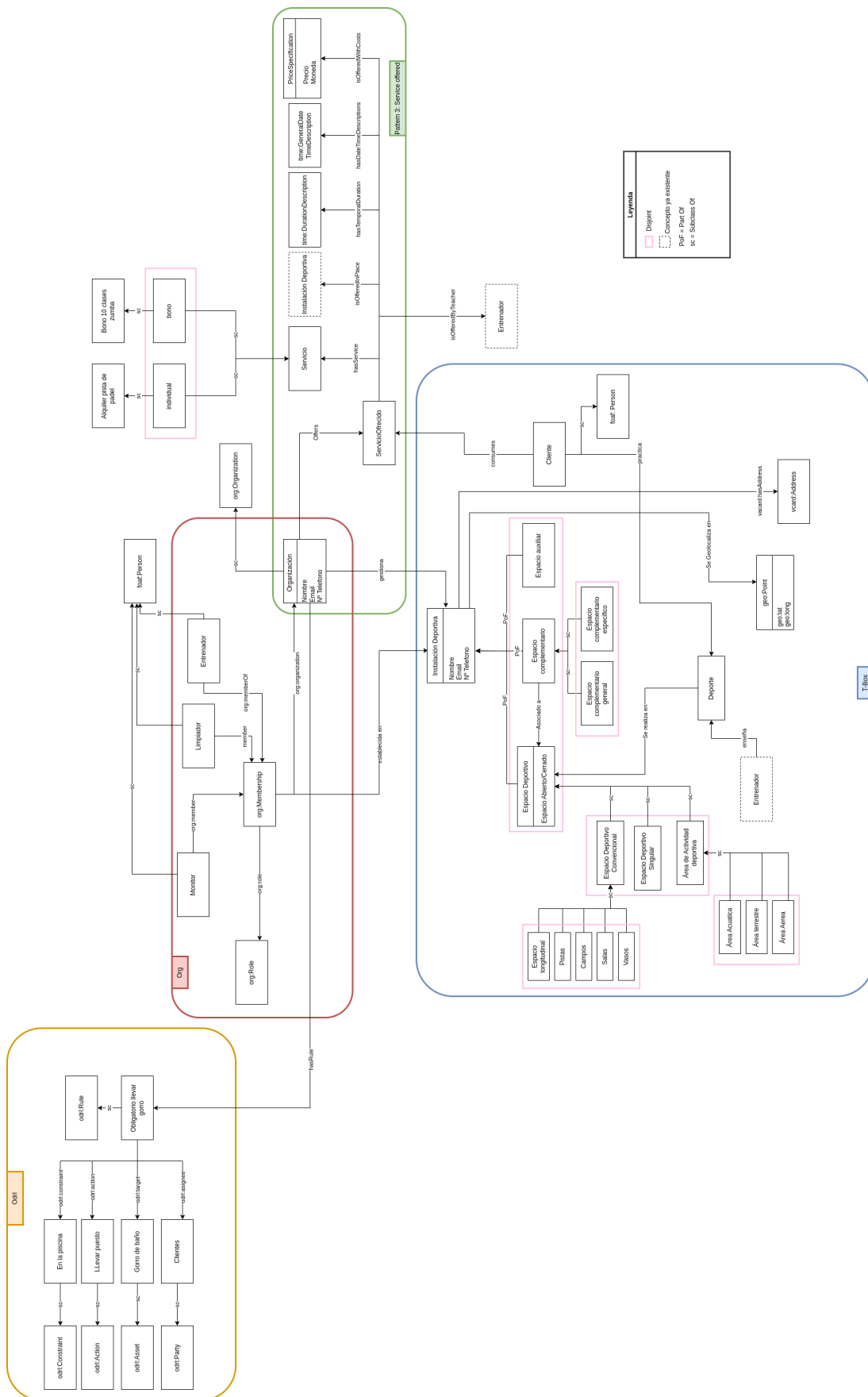


Figure 14: Diagrama completo del modelado de la ontología.



It is obvious that not all the pitfalls are equally important, their impact in the ontology will depend on multiple factors. For this reason, each pitfall has an importance level attached indicating how important it is. We have identified three levels:

- Critical** 🚨 : It is crucial to correct the pitfall. Otherwise, it could affect the ontology consistency, reasoning, applicability, etc.
- Important** 🟡 : Though not critical for ontology function, it is important to correct this type of pitfall.
- Minor** 🟢 : It is not really a problem, but by correcting it we will make the ontology nicer.

**Results for PQ4: Creating unconnected ontology elements.** 4 cases | Minor

Ontology elements (classes, object properties and datatype properties) are created isolated, with no relation to the rest of the ontology.

- The `pidtl` appears in the following elements:
  - <http://www.w3.org/ns/hingrade>
  - <http://schema.org/action>
  - <http://www.w3.org/ns/hingraphot>
  - <http://w3id.org/def/mailing-onto/instabclasses-deportvistiGeolocalizacion>

[illegible]

- The `ptid` appears in the following elements:
  - <http://www.ncbi.org/2008/mesh/daDxOntology>
  - <http://www.ncbi.org/2008/mesh/daTime>
  - <http://www.ncbi.org/2008/mesh/daDxOntologyDescription>
  - <http://www.ncbi.org/2008/cardi/mesh/daDxOntology>
  - <http://www.ncbi.org/2008/cardi/mesh/daTelephone>
  - <http://www.ncbi.org/2008/mesh/daDxOntology>
  - <http://www.ncbi.org/2008/mesh/daDxOntology>
  - <http://www.ncbi.org/2008/mesh/daDxOntology>
  - <http://www.ncbi.org/2008/mesh/daDxOntology>

[illegible]

**Results for P19: Defining multiple domains or ranges in properties.** 2 cases / Critical

The domain or range (or both) of a property (relationships and attributes) is defined by stating more than one class and/or equivalent statements. In OWL, identical or disjoint ranges are not allowed, but they are interpreted as conjunction, resulting in equivalent to the constant class intersection. This pitfall is related to the common error that appears when defining domains and ranges described in [7].

- This pitfall appears in the following elements:
  - 1 <http://www.w3.org/2009/09/meshhasTemporalDuration>
  - 2 <http://www.w3.org/2009/09/meshhasDate+TimeDescription>

**Results for P20: Missing ontology annotations.** 2 cases | Minor

The contents of some annotation properties are swapped or reused. This might affect minor annotation properties related to natural language information. The results might also be affected by the way the data is loaded or by providing descriptions such as `rdf:comment`. Other types of annotations could also be affected as temporal, versioning information, among others.

• This profile appears in the following elements:

- <http://www.w3.org/ns/holologue/organization>
- <http://www.w3.org/ns/holologue/hasMember>

Results for P22: Using different naming conventions in the ontology. ontology\* | Minor

The ontology elements are not named following the same convention (for example CamelCase or use of delimiters as "-" or "\_"). Some notations about naming conventions are provided in [2].

\*This pitfall applies to the ontology in general instead of specific elements.

Results for P24: Using recursive definitions. 2 cases | Important

An analogy relation (a class, an object property or a datatype property) is used in its own definition. Some examples of this would be: (a) the definition of a class as the enumeration of several classes including itself; (b) the appearance of a class while in its `equivalentClass` or `reflexiveClass` axioms; (c) the appearance of an object property in its `domain` or `range` `reflexiveRange` definitions; or (d) the appearance of a datatype property in its `range` domain definition.

• This pitfall appears in the following elements:

- <http://www.w3.org/2009/09/rdf-schema>

**Results for P29: Defining wrong transitive relationships.** 4 cases | Critical 

A relationship is defined as transitive, using `TransitiveProperty`, when the relationship is not necessarily transitive.

• This pitfall appears in the following elements:

- <http://www.w3.org/ns/aiid2/instance1>
- <https://schema.org/priceSpecification/hvacService>
- <https://schema.org/priceSpecification/maintenance>

**Results for P41: No license declared.** ontology\* | Important

The ontology metadata omits information about the license that applies to the ontology.

\*This pitfall applies to the ontology in general instead of specific elements.

**Suggestion:** symmetric or transitive object properties.

The domain and range assertions can be used for each of the following object properties. Could they be symmetric or transitive?

- <http://www.w3.org/2000/09/membership>
- <http://www.w3.org/2000/09/membership/Overlap>
- <http://www.w3.org/2000/09/membership/Disjoint>
- <http://www.w3.org/2000/09/membership/Member>
- <http://www.w3.org/2000/09/membership/Starts>
- <http://www.w3.org/2000/09/membership/Contains>
- <http://www.w3.org/2000/09/membership/Equals>
- <http://www.w3.org/2000/09/membership/Disjointly>
- <http://www.w3.org/2000/09/membership/Orderless>
- <http://www.w3.org/2000/09/membership/After>
- <http://www.w3.org/2000/09/membership/Before>
- <http://www.w3.org/2000/09/membership/Revolving>
- <http://www.w3.org/2000/09/membership/Identically>
- <http://www.w3.org/2000/09/membership/Subject>
- <http://www.w3.org/2000/09/membership/Objects>
- <http://www.w3.org/2000/09/membership/HasPart>

According to the highest importance level of pitfall found in your ontology the conformance badge suggested is "Critical pitfall" (see below). You can use the

It is obvious that not all the pitfalls are equally important; their impact in the ontology will depend on multiple factors. For this reason, each pitfall has an importance level attached indicating how important it is. We have identified three levels:

- Critical** 🚨: It is crucial to correct the pitfall. Otherwise, it could affect the ontology consistency, reasoning, applicability, etc.
- Important** ⚠️: Though not critical for ontology function, it is important to correct this type of pitfall.
- Minor** 🟡: It is not really a problem, but by correcting it we will make the ontology nicer.

**Results for P04: Creating unconnected ontology elements.** 2 cases | Minor

Ontology elements (classes, object properties and datatype properties) are created isolated, with no relation to the rest of the ontology.

- This pitfall appears in the following elements:
  - <http://schema.org/Action>
  - <http://www.w3.org/2013/org#Post>

**Results for PDR: Missing annotations.** 13 cases | [Minor](#) | [Major](#)

This pilot consists in creating an ontology element and failing to provide human readable annotations attached to it. Consequently, ontology elements lack annotation properties that label them (e.g. `rdfs:label`, `lexicon:LexicalEntry`, `skos:prefLabel` or `skos:altLabel`) or that define them (e.g. `rdfs:comment` or `descriptions`). This pilot is related to the guidelines provided in [5].

- The following elements have no `rdfs:label` defined:
  - <http://www.evo.wi.org/ontology/evost>
  - <http://www.w3.org/ns/ontology-wiki>
  - <http://icschema.org/Action>
  - <http://leximls.com/foaf/3person>
  - <http://www.w3.org/ns/ontology-Organization>
  - <http://leximls.com/foaf/3Ulgag>
  - <http://www.w3.org/2000/vcard/vcardpostal-code>
  - <http://icschema.org/ontology/ontologyCardinalTelephoneNumber>
  - <http://www.w3.org/2000/vcard/vcardnationality>
  - <http://www.w3.org/2000/vcard/vcardregion>
  - <http://www.w3.org/2000/vcard/vcardstreet-address>
  - <http://icschema.org/ontology/ontologyCardinalEmail>
  - <http://www.w3.org/2000/vcard/vcardcountry-name>

**Results for P11: Missing domain or range in properties.** 8 cases | Important

Object and/or domain properties without domain (or range of them) are included in the ontology.

This pitfall appears in the following elements:

- [http://www.w3.org/2000/01/rdf-schema#Duration](#)
- [http://www.w3.org/2000/01/rdf-schema#Time](#)
- [http://www.w3.org/2000/01/rdf-schema#Description](#)
- [http://www.w3.org/2000/01/rdf-schema#Email](#)
- [http://www.w3.org/2000/01/rdf-schema#Telephone](#)
- [http://www.w3.org/2000/01/rdf-schema#Day](#)
- [http://www.w3.org/2000/01/rdf-schema#Year](#)
- [http://www.w3.org/2000/01/rdf-schema#Month](#)

\* Tip: Solving this problem may lead to new results for other pitfalls and suggestions. We encourage you to solve all cases when needed and see what else you can get from OOPS!

---

**Results for P13: Inverse relationships not explicitly declared.** 45 cases | Minor

This pitfall appears when any relationship (except for those that are defined as symmetric properties using `SymmetricProperty`) does not have an inverse relationship (`owl:inverseOf` defined within the ontology).

\* OOPS! has the following suggestions for the relationships without inverse:

- `http://www.w3.org/2006/interva#interval` could be inverse of `http://www.w3.org/2006/interva#intervalDisjoint`
- `http://www.w3.org/2006/interva#intervalDisjoint` could be inverse of `http://www.w3.org/2006/interva#intervalAllPairs`

[illegible]

**Results for P20: Missing ontology annotations.** 2 cases | Minor

The contents of some annotation properties are swapped or mirrored. This pitfall might affect annotation properties related to natural language information (for example, annotations for naming such as rdfs:label or for providing descriptions such as rdfs:comment). Other types of annotation could also be affected as temporal, versioning information, among others.

• This pitfall appears in the following elements:

- <http://www.w3.org/ns/org/organization>
- <http://www.w3.org/ns/org/hasMember>

**Results for P22: Using different naming conventions in the ontology.** ontology\* | Minor

The ontology elements are not named following the same convention (for example CamelCase or use of delimiters as "-" or "\_"). Some notions about naming conventions are provided in [2].

\*This pitfall applies to the ontology in general instead of specific elements.

Results for P24: Use recursive definitions. 2 cases | Important

An ontology element (a class, an object property or a datatype property) is used in its own definition. Some examples of this would be: (a) the definition of a class as the enumeration of several classes including itself; (b) the appearance of a class within its `owl:equivalentClass` or `owl:disjointClass` axioms; (c) the appearance of an object property in its `owl:domain` or range `owl:range` definitions; or (d) the appearance of a datatype property in its `owl:domain` definition.

- This pitfall appears in the following elements:
  - <http://www.w3.org/2006/08/owlent.owl>
  - <http://www.w3.org/2009/09/owlent.owl>

**Results for P41: No license declared.** ontology" | Important

The ontology metadata omits information about the license that applies to the ontology.

\*This pitfall applies to the ontology in general instead of specific elements.

SUGGESTION: symmetric or transitive object properties.	16 cases
The domain and range axioms are equal for each of the following object properties. Could they be symmetric or transitive?	
<ul style="list-style-type: none"> <li><a href="http://www.w3.org/2000/10/interval/MeBy">http://www.w3.org/2000/10/interval/MeBy</a></li> <li><a href="http://www.w3.org/2000/10/interval/Overlaps">http://www.w3.org/2000/10/interval/Overlaps</a></li> <li><a href="http://www.w3.org/2000/10/interval/In">http://www.w3.org/2000/10/interval/In</a></li> <li><a href="http://www.w3.org/2000/10/interval/Disjoint">http://www.w3.org/2000/10/interval/Disjoint</a></li> <li><a href="http://www.w3.org/2000/10/interval/HasR">http://www.w3.org/2000/10/interval/HasR</a></li> <li><a href="http://www.w3.org/2000/10/interval/Starts">http://www.w3.org/2000/10/interval/Starts</a></li> <li><a href="http://www.w3.org/2000/10/interval/Equals">http://www.w3.org/2000/10/interval/Equals</a></li> <li><a href="http://www.w3.org/2000/10/interval/OverlapsBy">http://www.w3.org/2000/10/interval/OverlapsBy</a></li> <li><a href="http://www.w3.org/2000/10/interval/IsBefore">http://www.w3.org/2000/10/interval/IsBefore</a></li> <li><a href="http://www.w3.org/2000/10/interval/IsAfter">http://www.w3.org/2000/10/interval/IsAfter</a></li> <li><a href="http://www.w3.org/2000/10/interval/IsDuring">http://www.w3.org/2000/10/interval/IsDuring</a></li> <li><a href="http://www.w3.org/2000/10/interval/IsInsideBy">http://www.w3.org/2000/10/interval/IsInsideBy</a></li> <li><a href="http://www.w3.org/2000/10/interval/IsBeforeOr">http://www.w3.org/2000/10/interval/IsBeforeOr</a></li> <li><a href="http://www.w3.org/2000/10/interval/IsAfterOr">http://www.w3.org/2000/10/interval/IsAfterOr</a></li> <li><a href="http://www.w3.org/2000/10/interval/IsIn">http://www.w3.org/2000/10/interval/IsIn</a></li> </ul>	

According to the highest importance level of pitfall found in your ontology the conformace badge suggested is "important pitfalls" (see below). You can use the