

Ontologías en la Web

Trabajo Práctico Final:

“Ontología de Turismo”

Autores:

Amorós, Juan Manuel

Catalano, Juan Ignacio

Pose, Alberto Miguel

Introducción

El objetivo del trabajo es la creación de una ontología para describir atracciones turísticas. La herramienta que se utilizó para crear la ontología fue Protegé, desarrollada por la universidad de Stanford para la creación de ontologías.

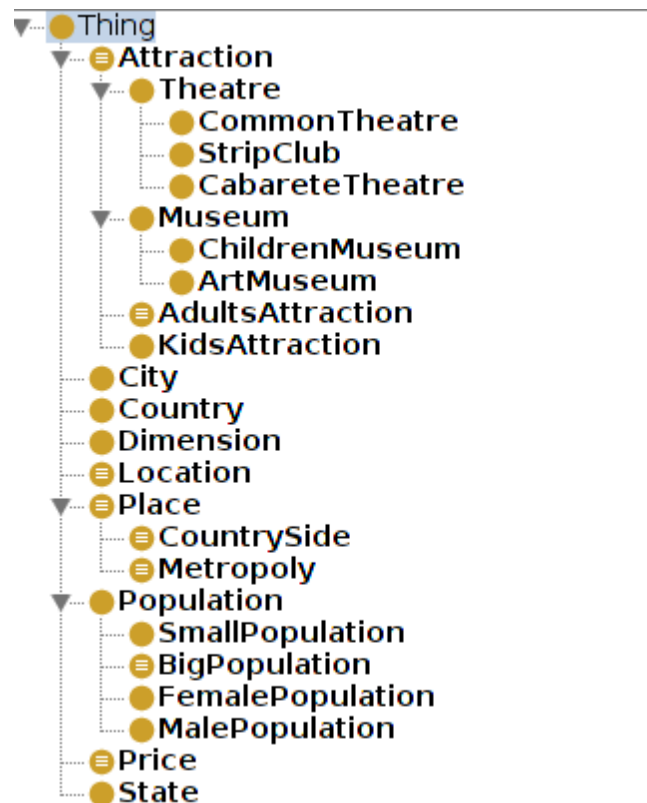
Desarrollo

Se presentan las consideraciones tomadas en cuestiones de diseño e implementación para facilitar el entendimiento de la ontología desarrollada.

Diseño

Clases

El diagrama de jerarquía de clases es el siguiente:



Como vemos en dicho diagrama, se definieron las siguientes clases:

- **Attraction.** Esta clase representa a las atracciones turísticas. Esta clase tiene las siguientes subclases:
 - **Theatre.** Como es evidente, esta clase representa los teatros, que pueden ser de 3 tipos:
 - **CommonTheatre.** Estos son los teatros comunes.
 - **StripClub.** Para mayores de 18.
 - **CabareteTheatre.** Para mayores de 21.
 - **Museum.** Esta clase define los museos que pueden ser de dos tipos:
 - **ArtMuseum.** Museos de arte.
 - **ChildrenMuseum.** Museos para niños, por ejemplo: el museo participativo.
 - **AdultsAttraction.** Representa a las atracciones que son solo para adultos.
 - **KidsAttraction.** Representa a las atracciones orientadas a niños.
- **Place.** Es la clase que representa a los lugares que pueden tener o no atracciones.
 - **CountrySide.** Aquellos lugares con poca cantidad de habitantes.
 - **Metropoly.** Son las grandes ciudades, con poblaciones mayores a 2.000.000 de personas.
- **City.** Representa una ciudad.
- **State.** Representa un estado que puede contener una o más ciudades.
- **Country.** Representa un país que puede tener uno o más estados.
- **Population.** Representa la población de un determinado Place.
 - **SmallPopulation.** Representa una población menor a 2.000.000 de habitantes.
 - **BigPopulation.** Representa una población mayor o igual a 2.000.000 de habitantes.
 - **MalePopulation.** Representa la cantidad de habitantes de sexo masculino.
 - **FemalePopulation.** Representa lo opuesto a lo anterior.
- **Price.** Representa el precio de las atracciones.

Propiedades de Objetos



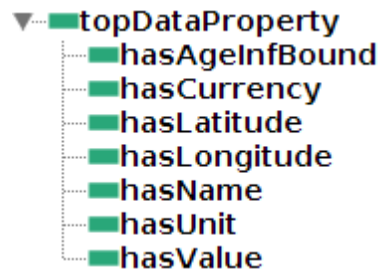
Como vemos en el diagrama tenemos las siguientes propiedades de objetos:

- **hasAttraction:** Asocia un lugar con una atracción turística (propiedad inversa:

isLocatedIn).

- hasCity: Está en una determinada ciudad.
- hasState: Está en determinado estado.
- hasCountry: Está en un determinado país.
- hasDimension: Ocupa una determinada superficie.
- hasLocation: Está ubicando en.
- hasPopulation: Tiene cierta cantidad de habitantes.
- hasPrice: Cuanto cuesta la atracción.

Propiedades de Datos



Estas se definen como:

- hasAgeInfBound: Límite inferior de edad.
- hasCurrency: Nombre de la moneda.
- hasLatitude: Latitud donde está ubicado.
- hasLongitude: Longitud donde está ubicado.
- hasName: Nombre.
- hasUnit: Unidad (por ejemplo, m²).
- hasValue: Valor.

Decisiones de Implementación

Clases Definidas

Las siguientes clases se decidió hacerlas definidas, esto es que sea condición necesaria y suficiente para pertenecer a la misma cumplir con las condiciones que la definen:

- Attraction
- AdultsAttraction
- Location
- Place
- CountrySide
- Metropoly
- BigPopulation
- Price.

Clases disjuntas

Las siguientes clases fueron definidas disjuntas entre si, debido a su naturaleza exclusiva:

- AdultsAttraction y KidsAttraction.
- CountrySide y Metropoly.
- SmallPopulation y BigPopulation.
- MalePopulation y FemalePopulation.

Axioma de cobertura

Se implementó este axioma para la clase Place, indicando que un individuo de la clase Place debe ser o una Metropoly o un CountrySide, es decir, un lugar en el campo.

Inferencias

Para mostrar las inferencias que se pueden realizar con Protegé, aprovechando todas las capacidades del razonador provisto por la herramienta, decidimos implementar diferentes restricciones en algunas de las clases de la ontología. Por ejemplo,

- Dadas las clases StripClub y CabareteTheatre como subclases de Theatre, el razonador puede inferir que además ambas son subclases de AdultsAttraction, ya que restringen los valores que sus individuos pueden tomar para la propiedad hasAgeInfBound de manera tal que resultan siendo atracciones para adultos.
- aFemalePopulation es un individuo que si bien es creado como instancia de la clase FemalePopulation, se infiere que es también miembro de la clase BigPopulation debido a su gran cantidad de habitantes.
- inferredKidsAttraction, es instanciado como una Attraction, pero al tener su propiedad hasAgeInfBound fijado en 9 años, se infiere que es una atracción para niños, es decir, es miembro de la clase KidsAttraction. Cabe destacar que la clase AdultsAttraction posee una restricción de cardinalidad en su propiedad hasAgeInfBound (≥ 12).
- BuenosAiresPlace se infiere como miembro de la clase Metropoly ya que contiene al menos una población que es instancia de BigPopulation.
- Desde ya que en el caso de ser instancias de subclases, los individuos también figuran como miembros de las superclases correspondientes, por ejemplo en el caso de aFemalePopulation y aMalePopulation.
- Se creó la propiedad transitiva isMorePopulatedThan que se encuentra definida para Paris y Roma de la siguiente forma:
 - Paris tiene mayor población que Roma.
 - Roma se encuentra más poblado que Buenos Aires.
 - De estas dos, el razonador infiere que Paris se encuentra más poblado que Buenos Aires.

Conclusiones

Utilizando las capacidades provistas por la herramienta Protegé para el diseño e implementación de ontologías, se pudo extender las funcionalidades deseadas para agregar contenido semántico al modelo de datos. El haber utilizado OWL permitió superar las limitaciones impuestas por RDF/RDF Schemma, y poder definir ciertas características como:

- La disjunción de clases.
- Restricciones de rangos.
- Características de las propiedades. Por ejemplo, la transitividad que definimos para `isMorePopulatedThan`.
- Restricciones de cardinalidad. Como por ejemplo la que se definió para la clase `AdultsAttraction`.
- Combinaciones booleanas de clases. Por ejemplo, la clase `Place` se definió como la unión de `CountrySide` y `Metropoly`.

Estas extensiones fueron aprovechadas para poder inferir información y extender la potencia semántica del modelo de datos turísticos.