Teleology Protégé Demo (iTelos Informal Modelling)

INPUT:

<u>OWL RDF/XML Files of Entity Types Produced During Inception (I2SF/Inception).</u> **Example-** OWL RDF/XML Files of each single dataset: *BusRoute, BusAgency, Park, BusStop, StudentResidence*

<u>Object Properties Informed By Competency Questions (Inception)</u> **Example- operates** as in *BusAgency - operates - BusRoute* [from the example CQ: "Which bus routes do Trentino Transporti SpA operate?"]

DURING INFORMAL MODELLING:

You Build The Visual Diagram of the Teleology (informed by the **INPUT** above; example (FIG.1) below):



Build the OWL RDF/XML axiomatization of the Teleology (such as the one above):

step (0): create a new entity type in Protégé with the following information (and save it!):

ds:Region (class) [with following data properties:]

- ds:regionLatitude
- ds:regionLongitude
- ds:regionAltitude
- ds:startTime
- ds:endTime

step (1): open a new ontology project in Protégé [File -> New]. Create a new class *ds:Everything*.

step (2): import each entity type files (OWL RDF/XML files) in the new ontology project

- Go to "Active Ontology" Tab
- Go to "Ontology imports" Tab
- Click "+" on "Direct Imports"
- Choose appropriate option and finish the import

[Repeat this for all relevant entity type files (all from the INPUT + ds:Region) required for your teleology]

[Verify whether all the relevant entity types have been properly imported]

step (3): create object properties (from now on, OP) using "Object Property" Tab (within the "Entities" Tab).

- All the white rhombus (e.g., operates, locatedIn) in the visual diagram of the teleology (FIG. 1) should be codified as OPs [NOTE: create only one OP in Protégé even if it is occurring multiple times in the teleology visual diagram.]
- TWO CASES:
 - If an OP occurs only once in the teleology, encode its domain and range etypes. Ex: in the FIG. 1 above, 'operates' is the OP with domain etype 'BusAgency' and range etype 'BusRoute'.



OR

 If an OP occurs multiple times in the teleology, it means it has potentially multiple etypes as domain and range. In that case, create the OP but don't encode its domain and range etypes.
 Instead, select each of the multiple (domain) etypes and indirectly enforce the domain-range via the object restriction creator.

EXAMPLE below (in sync with FIG. 1):

near is an OP which occurs in *park - near - BusStop* as well as *StudentResidence - near - BusStop*. In this case, *near* is the same OP having two different domain etypes *park* and *StudentResidence* and the same range, i.e. *BusStop*. In this case,

Select *park* (which is the domain etype) in the class hierarchy, click on '+' in SubClass Of, then select 'Object restriction creator' and then assert the OP as well as the range. Keep the restriction type as 'Some (existential)'. Do the same for *StudentResidence*.

🍕 untitled-ontology-391 [http://www.semanticweb.org/mbagc/ontologies/2022/10/untitled-ontology-391) : [http://www.semanticweb.org/mbagc/ontologies/2022/10/untitled-ontology-391] 🧧 📙 X				
File Edit View Reasoner Tools Refactor Window Help				
<pre>< > * untitled-ontology-391 (http://www.semanticwe</pre>	b.org/mbagc/ontologies/20	022/10/untitled-ontology-391)	~ Search	
Active ontology × Entities × Individuals by class × DL Que	$rv \times OntoGraf \times$			
Annotation properties Datatypes Individuals	e park — http://www.co	o-ode.ora/ontologies/ont.owl#GO_6000000		
Classes Object properties Data properties	Annotations Usage			
Class hierarchy: park	Annotations: park			
owl:Thing	Annotations		000	
BusStop post	park			
	Description: park	🔏 park 🛛 🕹 🖌		
	Equivalent To 🕂	Data restriction creator Class hierarchy		
	SubClass Of 🕂	Object restriction creator Class expression editor		
	Canaral alaga aviama	Restricted property Restriction filler		
	General class axions	Asserted V Asserted V Asserted V		
	SubClass Of (Anonymous	e owi:topObjectProperty e owi: I hing near eBusStop		
	Instances 🕂	-• park		
	Target for Key			
	raiget for Key	Restriction type		
	Disjoint With 🕂	Some (existential) Cardinality 1		
	Disjoint Union Of 🕂	OK Canaal		
		UK Cancel	Reasoner > Start reasoner 🔽 Show inferences 📳	
۵ کې 🛃	arch 🔲 🗩 🧳 🍕	" 🔁 🔽 🔕 🔣 🔄 🗠 🛱 🕅	令 ↓) 15:38 15-11-2022 4	

This will create a property constraint : *park - near - some BusStop* which will indirectly axiomatize the domain and range of the OP 'near'. Do this for all OPs which occur multiple times in the teleology.

🝕 untitled-ontology-391 (http://www.semanticweb.org/mbagc/ontologies/2022/10/untitled-ontology-391) : [http://www.semanticweb.org/mbagc/ontologies/2022/10/untitled-ontology-391] – 🛛				
File Edit View Reasoner Tools Refactor Window Help				
> wuntitled-ontology-391 (http://www.semanticweb. View	.org/mbagc/ontologies/2022/10/untitled-ontology-391)	Search		
Active ontology \times Entities \times Individuals by class \times DL Query	$\prime \times \text{OntoGraf} \times$			
Annotation properties Datatypes Individuals	e park — http://www.co-ode.org/ontologies/ont.owl#GO_6000000			
Classes Object properties Data properties	Annotations Usage			
😘 🛻 🕺 🛛 Asserted 🗸	Annotations +			
• owl:Thing • BusStop	rdfs:label [language: en]	@×O		
- ● park	park			
	Description: park			
	Equivalent To 🛨			
	SubClass Of 🔒			
	•near some BusStop	?@XO		
	General class axioms 🕂			
	SubClass Of (Anonymous Ancestor)			
	Instances 🕂			
	Target for Key 🕂			
	Disjoint With			
		-		
To use the reasonse' black Reasonse''s State Reasonse'''s State Reasonse'''s State Reasonse'''s State Reasonse'''s State Reasonse'''s State Reasonse'''''''''''''''''''''''''''''''''''				
Search) 🗖 🖻 🧳 🚰 🗾 🔽 S K 🖪 🖳 🗠 🛭 🕬	奈 (41)) 🗗 15:42 👍 15-11-2022 🔮		

step (4): finally, after encoding the entire teleology following the above steps, save it in OWL RDF/XML format. You have your teleology axiomatized in Protégé !!!