The One Piece Ontology

-Protégé Application

KIRAN TEJA KOLLI

INFO 6540 Data Management

Instructor: Elvira Mitraka

Date: 6th March, 2018

DALHOUSIE UNIVERSITY

The protégé software is open source platform that provides a user community with tools to build domains and knowledge-based models with ontologies. The software is used to build different ontologies with adding various attributes. It describes the classes created and its relations by object properties. This software helps the user to create an ontology with different classes and with multiple object properties. Each class created can be defined using an annotation called "isDenfiendBy".

In this report, the creation of "One Piece" ontology is discussed and relation between each class is described. The One Piece ontology states the different classes that are created under the parent class- One Piece. The Annotation properties are set to The sub classes for the parent class are:

1. Characters.
2. Cast and Crew.
3. Devil fruit users.

The figure 1.a shows the main class-"OnePiece and the 3 sub classes.

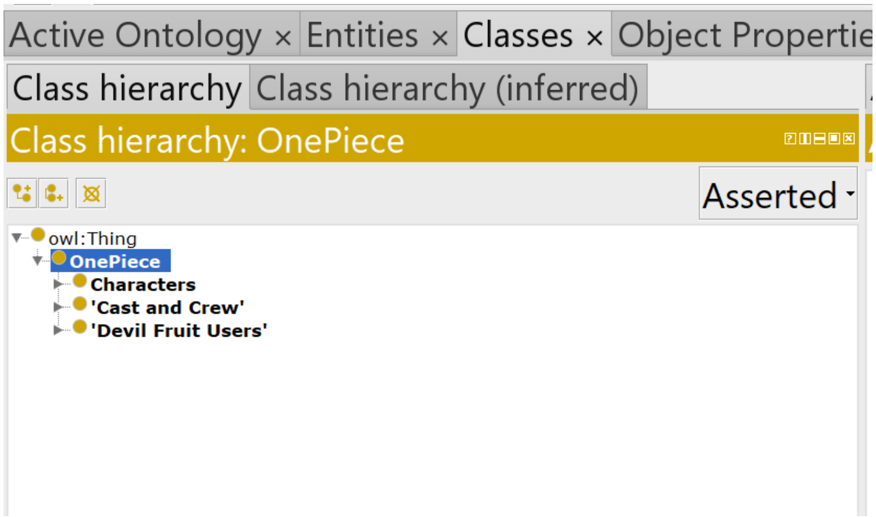
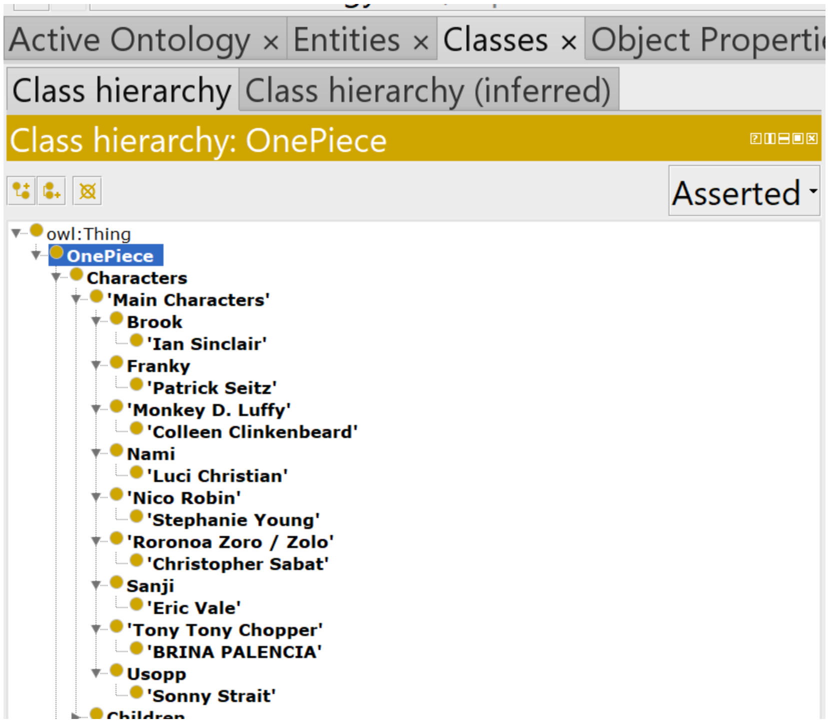


Figure: 1.a

The first sub class is "characters" which has two child classes:1-Main Characters 2-Children. This classes are created to differentiate the parent class called "characters". The Main Characters class is created using the creation of class option in the Classes tab in Protégé. This class mainly describes about the special and main role members of One Piece series. One object property is created which is named as "voiceGivenBy", this object property is defined as the relation between main animated character and the real person who gave his voice over. There is one more sub class created for each main character which holds the label of real time person. The relation between these both classes will be defined by using an object property. Using this object property, it states that the main characters voice over is given by the child class label.



```
owl:Thing
  OnePiece
    Characters
      'Main Characters'
        Brook
          'Ian Sinclair'
        Franky
          'Patrick Seitz'
        'Monkey D. Luffy'
          'Colleen Clinkenbeard'
        Nami
          'Luci Christian'
        'Nico Robin'
          'Stephanie Young'
        'Roronoa Zoro / Zolo'
          'Christopher Sabat'
        Sanji
          'Eric Vale'
        'Tony Tony Chopper'
          'BRINA PALENCIA'
        Usopp
          'Sonny Strait'
      Children
```

Figure 2.a

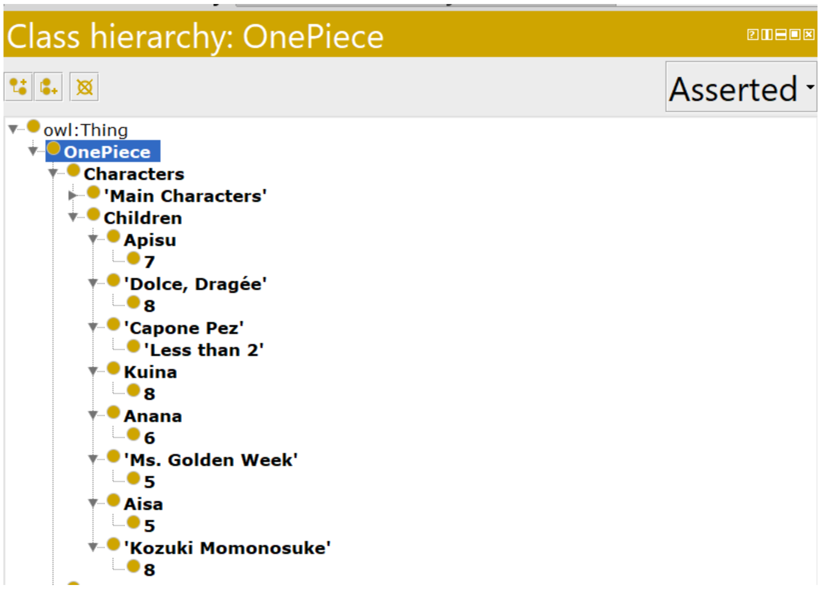
In figure 2.a, Brook is one of the main character of the series and "Ian Sinclair" is the person who gave voice to the character in animated series. Similarly, Franky is the animated series and Patrick Seitz is the person who gave voice over to the character. This relation is defined by using object property that is named as "voiceGivenBy".

The reverese methodology states another object property that is "givenVoiceTo". This object property is also between the main character and person who gave voice over. But this time, the relation is given from the person to character i.e., 'Ian Sinclair' to 'Brook'. It is represented as

'Ian Sinclair'---(has given voice to)-→ 'Brook'. This is the second object property that is been used in the ontology.

The second subclass of Characters class is "Children". This class gives the details such as age of the child characters that are involved in the animated series. In Figure 3.a, the parent class is children and the subclasses are names of child characters. The age groups are subclasses of names of child characters. Therefore the new object property has been created such as "isInAgeGroupOf". This object property gives the age category relation for both name of child character and age of the character.

E.g., Apisu---→(is in age group of)----→8 years.

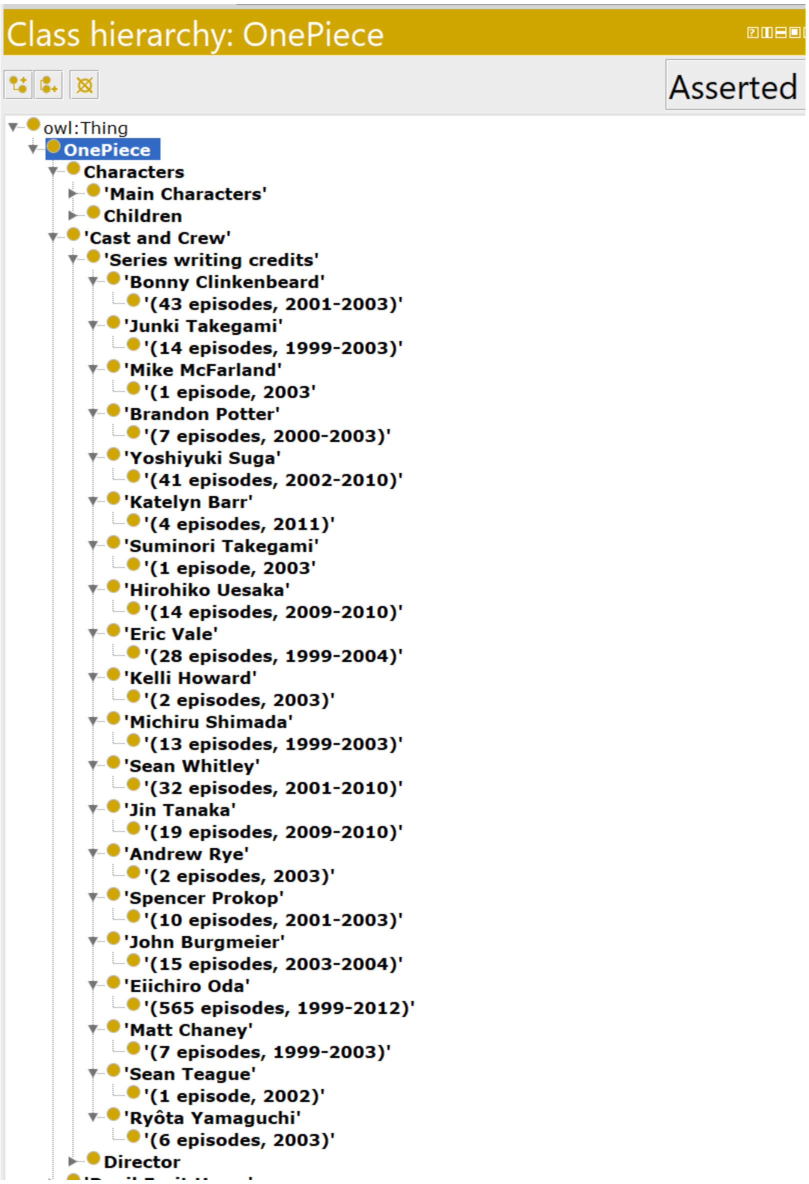


- owl:Thing
  - OnePiece
    - Characters
      - 'Main Characters'
      - Children
        - Apisu
          - 7
        - 'Dolce, Dragée'
          - 8
        - 'Capone Pez'
          - 'Less than 2'
        - Kuina
          - 8
        - Anana
          - 6
        - 'Ms. Golden Week'
          - 5
        - Aisa
          - 5
        - 'Kozuki Momonosuke'
          - 8

Figure 3.a

The “Children” class is subclass of “characters”. The names such as Apisu, Kuina, Anana act as subclasses for the parent class-“Children” and also act as parent classes for the age groups.

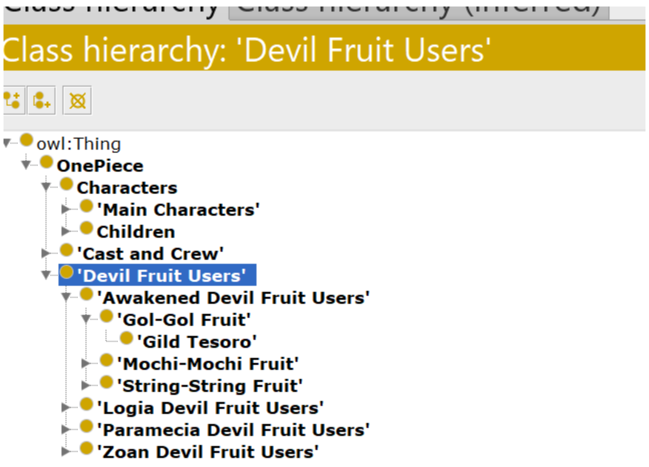
Similarly the second domain created is about cast and crew class which is a subclass of One piece. This class involves the characters who worked for the animated series and who has directed the episodes. The Directors and Writers of the series are created as subclasses for this parent class. The figure 4.a states that Director and Series writing credits are sub classes. The following children classes are created with relations named as ‘isWrittenBy’ and ‘directedBy’. The object properties are used as connectors between the person name and episodes written by them. Similarly, the ‘directedBy’ property is used to relate the episodes directed by respective director.



```
owl:Thing
  OnePiece
    Characters
      'Main Characters'
      Children
    'Cast and Crew'
      'Series writing credits'
        'Bonny Clinkenbeard'
          '(43 episodes, 2001-2003)'
        'Junki Takegami'
          '(14 episodes, 1999-2003)'
        'Mike McFarland'
          '(1 episode, 2003'
        'Brandon Potter'
          '(7 episodes, 2000-2003)'
        'Yoshiyuki Suga'
          '(41 episodes, 2002-2010)'
        'Katelyn Barr'
          '(4 episodes, 2011)'
        'Suminori Takegami'
          '(1 episode, 2003'
        'Hirohiko Uesaka'
          '(14 episodes, 2009-2010)'
        'Eric Vale'
          '(28 episodes, 1999-2004)'
        'Kelli Howard'
          '(2 episodes, 2003)'
        'Michiru Shimada'
          '(13 episodes, 1999-2003)'
        'Sean Whitley'
          '(32 episodes, 2001-2010)'
        'Jin Tanaka'
          '(19 episodes, 2009-2010)'
        'Andrew Rye'
          '(2 episodes, 2003)'
        'Spencer Prokop'
          '(10 episodes, 2001-2003)'
        'John Burgmeier'
          '(15 episodes, 2003-2004)'
        'Eiichiro Oda'
          '(565 episodes, 1999-2012)'
        'Matt Chaney'
          '(7 episodes, 1999-2003)'
        'Sean Teague'
          '(1 episode, 2002)'
        'Ryôta Yamaguchi'
          '(6 episodes, 2003)'
      Director
```

Figure 4.a

The last class with URI: OnePiece0000003 is created and names as “Devil Fruit users”. This class is defined as the special creatures that are introduced in the series. These fruits are consumed by multiple characters in the series but we are using ‘ONLY’ as an attribute in the object property tab because each fruit is consumed only by one character. Hence it’s a direct relation between fruit and character.



Class hierarchy: 'Devil Fruit Users'

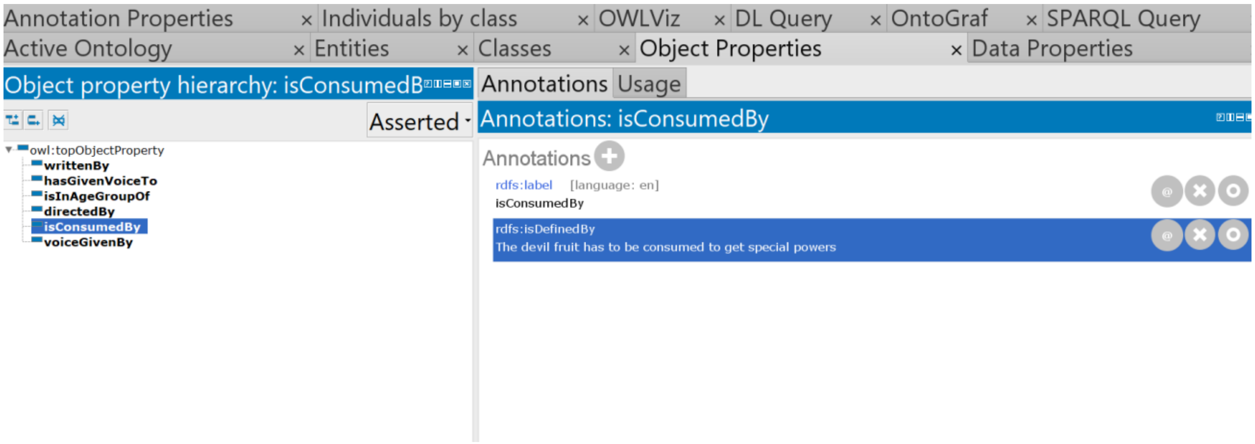
- owl:Thing
  - OnePiece
    - Characters
      - 'Main Characters'
      - Children
    - 'Cast and Crew'
    - 'Devil Fruit Users'
      - 'Awakened Devil Fruit Users'
        - 'Gol-Gol Fruit'
          - 'Gild Tesoro'
        - 'Mochi-Mochi Fruit'
        - 'String-String Fruit'
      - 'Logia Devil Fruit Users'
      - 'Paramecia Devil Fruit Users'
      - 'Zoan Devil Fruit Users'

Figure 6.a

In Figure 6.a, 'Awakened' is a category of Devil fruit, so it is created as subclass of "Devil fruit users". All the categories of Devil fruits are created and each one has its owner which is defined by an object property called "hasConsumedBy".

Gol-Gol Fruit ----->(isConsumedBy)------>'Gild Tesoro'.

Similarly fruits in different categories are related with the owners respectively with the same object property.



Object property hierarchy: isConsumedBy

- owl:topObjectProperty
  - writtenBy
  - hasGivenVoiceTo
  - isInAgeGroupOf
  - directedBy
  - isConsumedBy
  - voiceGivenBy

Annotations: isConsumedBy

Annotations

rdfs:label [language: en]
isConsumedBy

rdfs:isDefinedBy
The devil fruit has to be consumed to get special powers

Figure7.a

In Figure 7.a, All the object properties that are created to define a relation between classes are stated. Each object property has its own definition of relation and used to relate between a class and subclass.

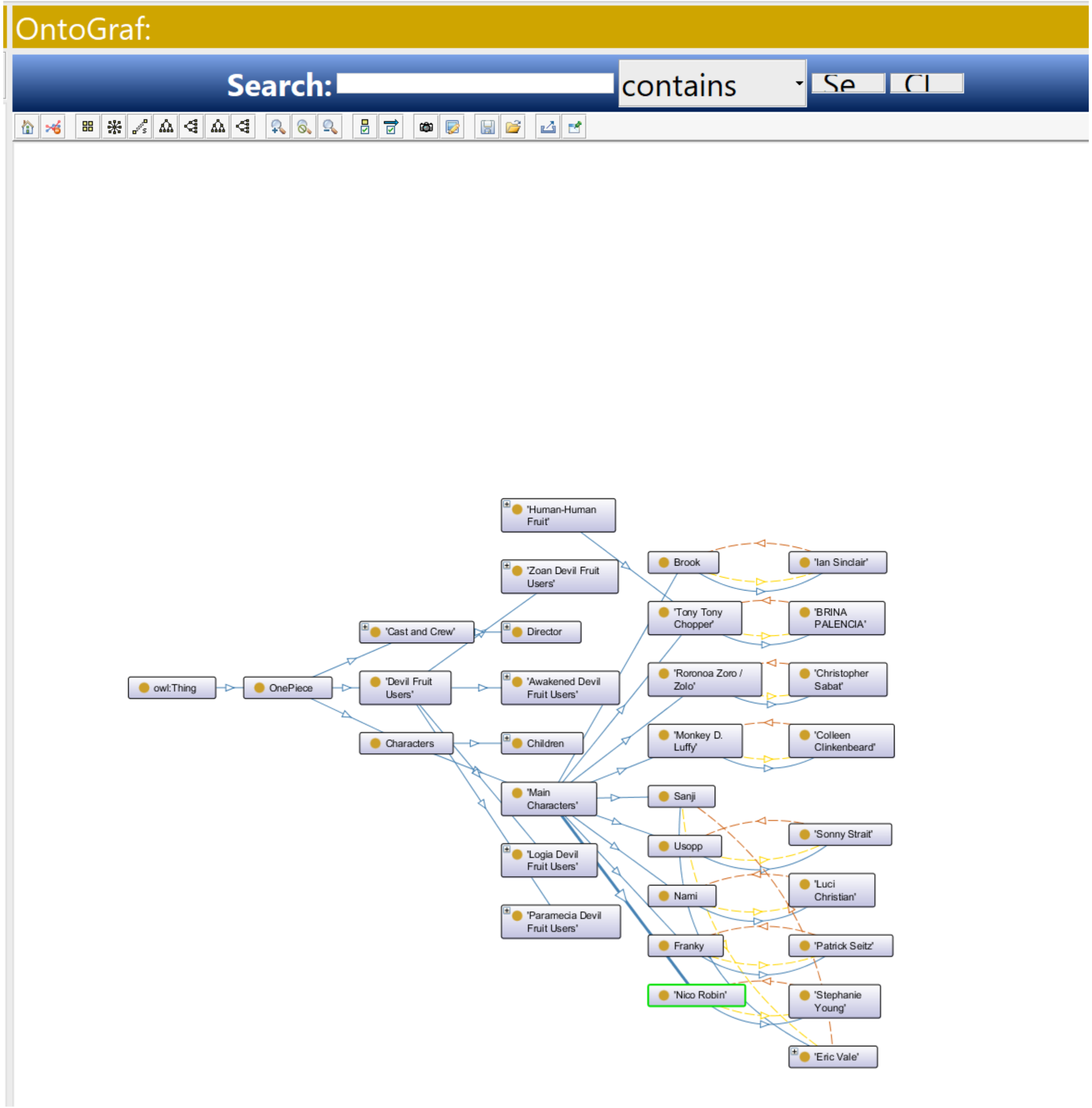


Figure 8.a OntoGraph

In the above Figure, a sample ontograph shows a relation between one set of class and subclass. The table expands shows all the relations between classes and subclasses that are created in the ontology. The tree structure will give multiple definitions and relations created between the classes and subclasses.

Learning outcome:

Ontology has given me the knowledge of creating a relation between two objects and defining them. Each class is created under which multiple subclasses can be fixtured. Using of annotation properties has given my ontology with a unique identifier and separate URI. The tutorial of protégé is very helpful and can give a clear picture of the application and will guide the user to create a basic ontology with multiple attributes.