Editing e ragionamento con Protégé

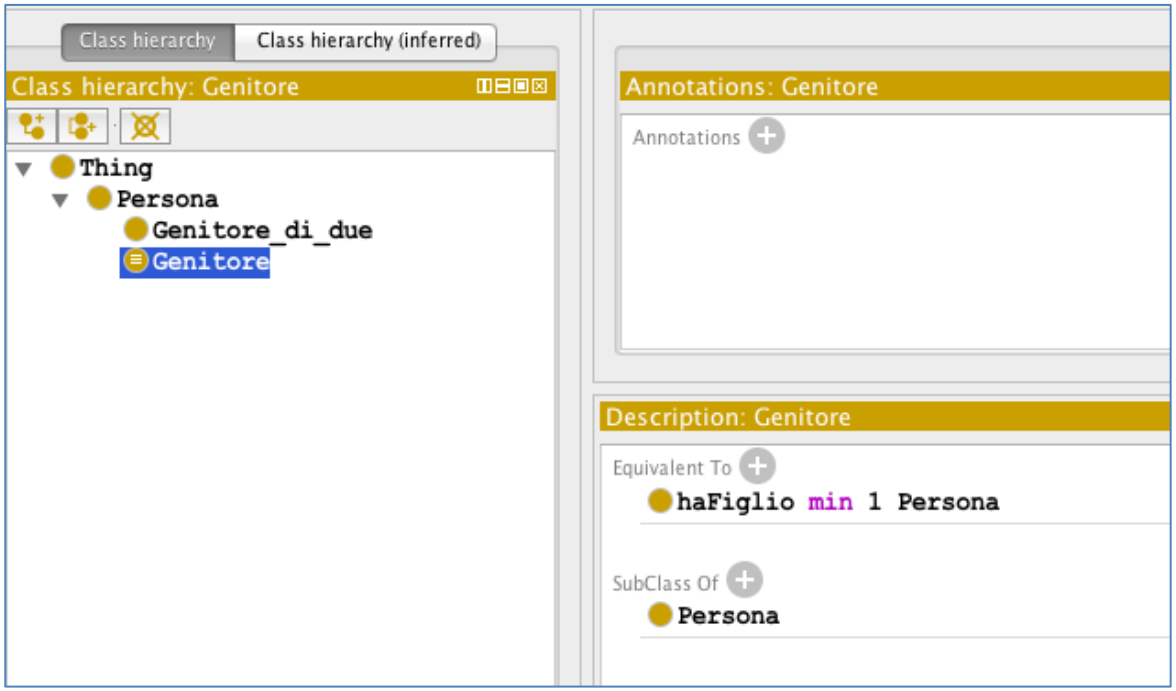
Strumenti di editing

Strumenti per il ragionamento automatico

Ragionamento su classi

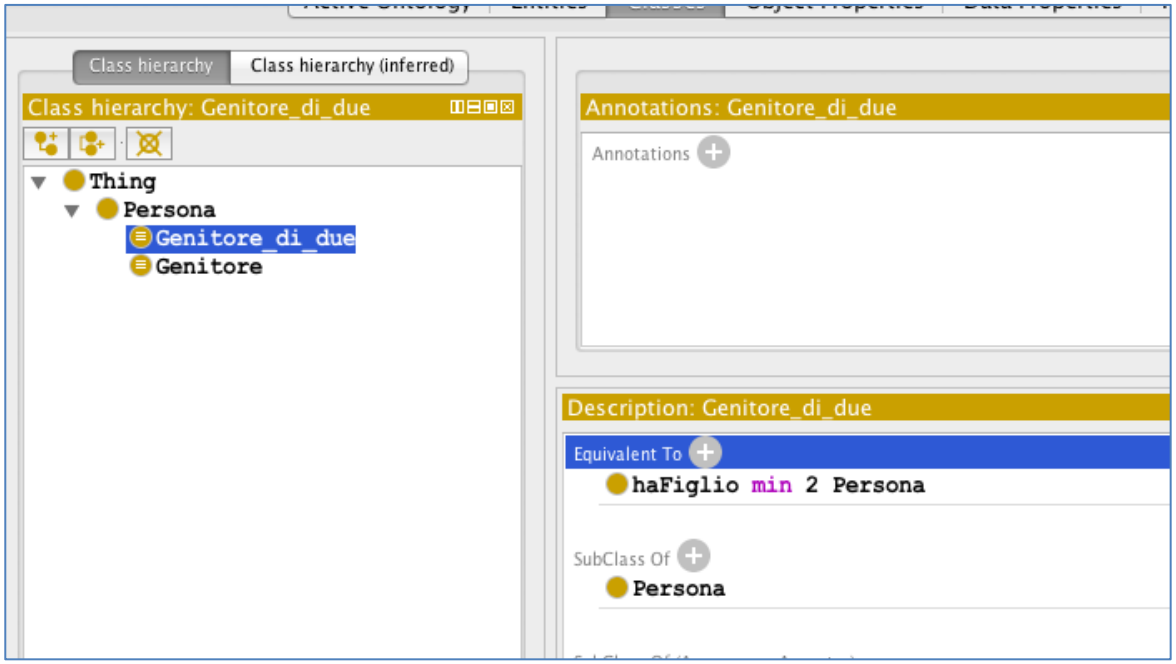
La gerarchia delle classi asserita mediante gli assiomi di sottoclasse può essere diversa da quella inferita

Il reasoner può inferire delle relazioni di *sussunzione* a partire dalla descrizione delle classi



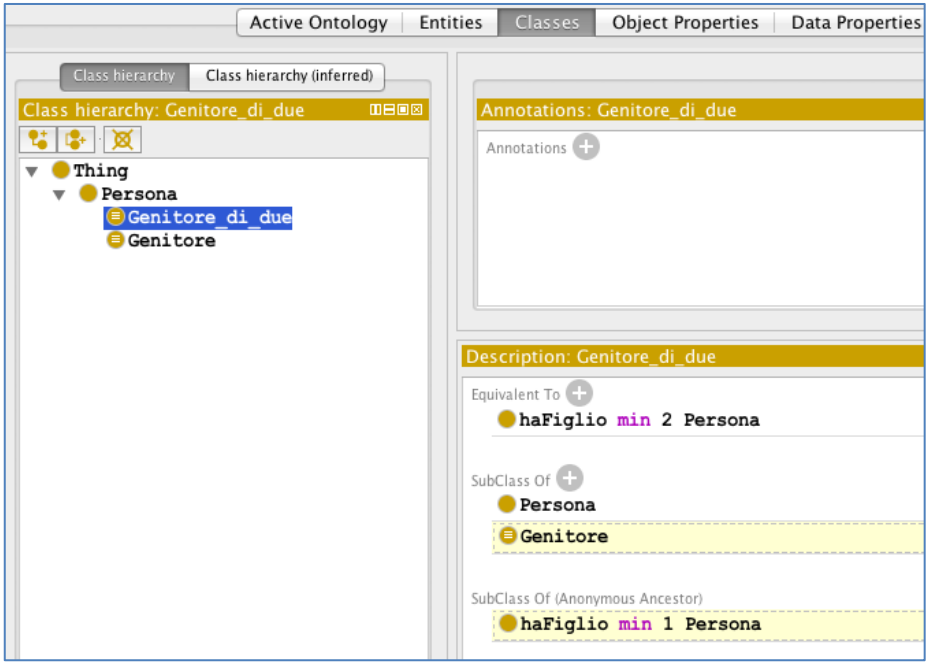
La proprietà haFiglio ha come dominio Persona e come range Persona

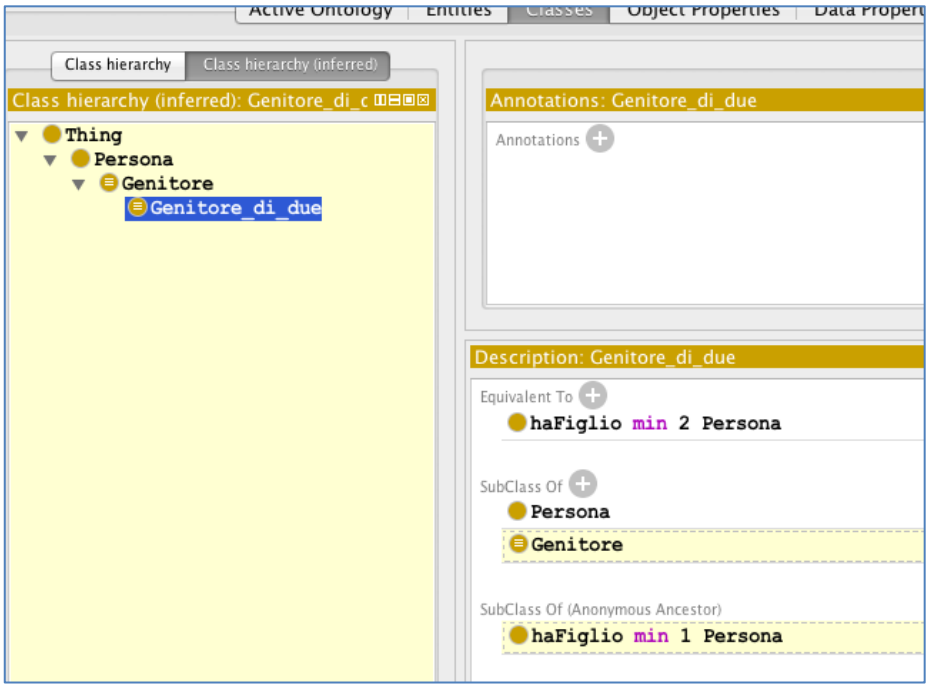
La classe Genitore è vincolata a avere almeno un figlio



La classe Genitore_di_due è vincolata a averne almeno due

Le due classi sono definite come sottoclassi di Persona.





Il reasoner inferisce che la classe Genitore_di_due è sottoclasse di Genitore

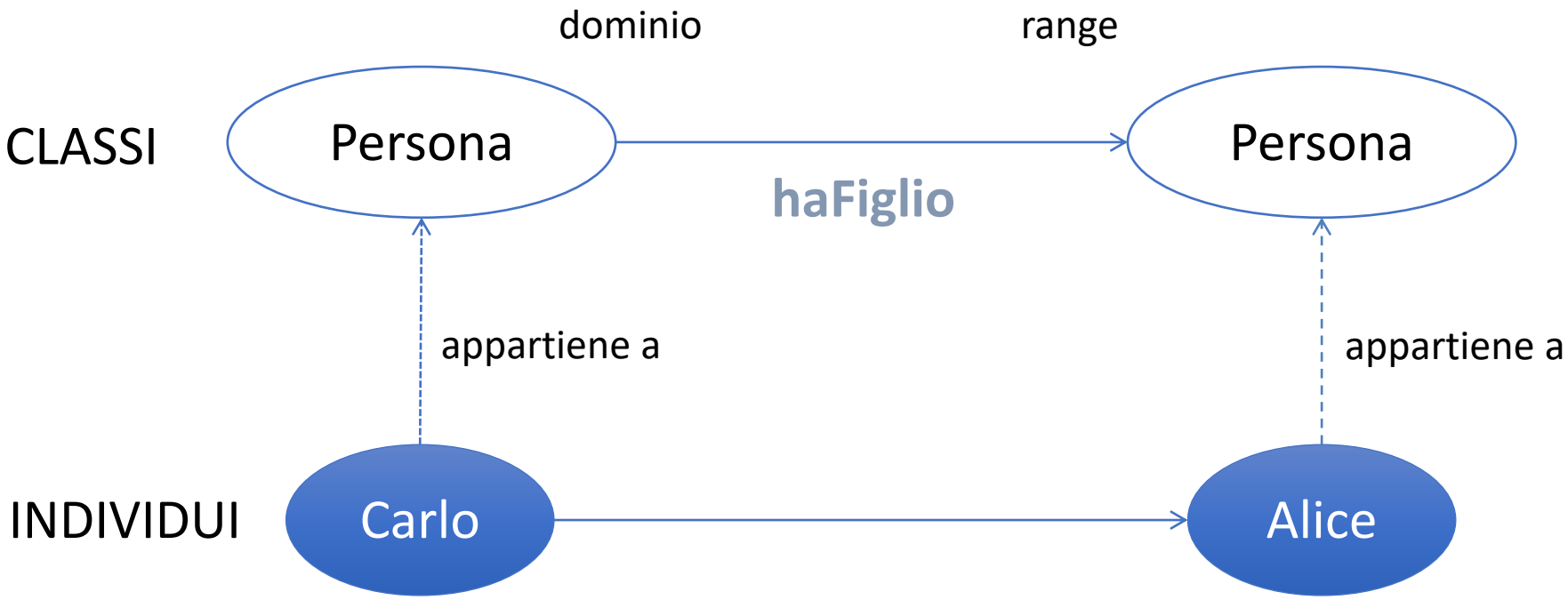
Nel pannello delle classi, la gerarchia inferita mostra che la classe Genitore_di_due è stata spostata sotto la classe Genitore

Il reasoner ha inferito la relazione di sussunzione

Inserimento di individui

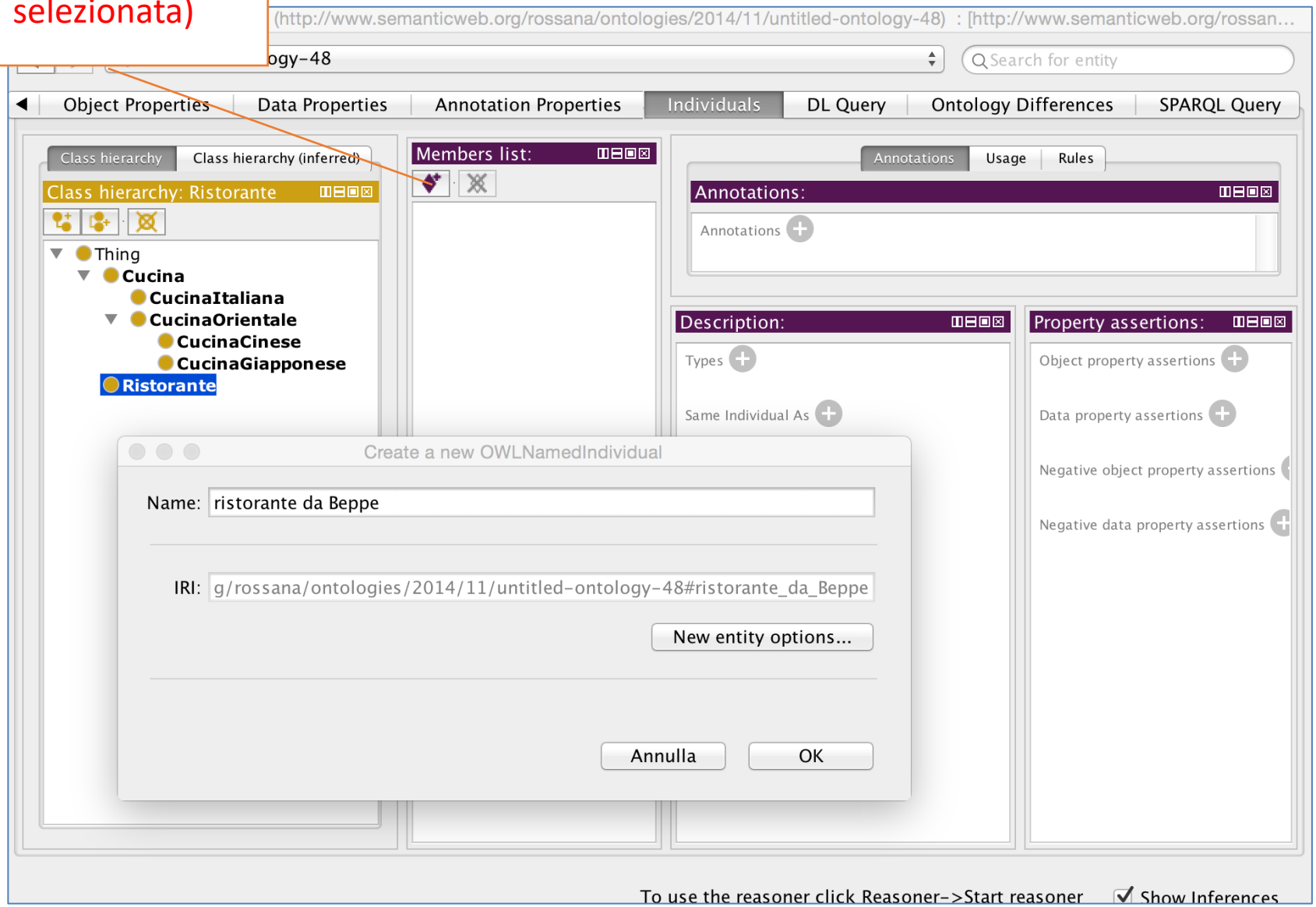
- Gli individui vengono inseriti direttamente in una classe
- Dopo averli inseriti è possibile predicarne le caratteristiche, cioè specificarne le proprietà che li mettono in relazione con altri individui o con un dato (class e property assertions)
- Il ragionamento può eventualmente ricollocare un certo individuo in un'altra classe

Relazione classi - individui

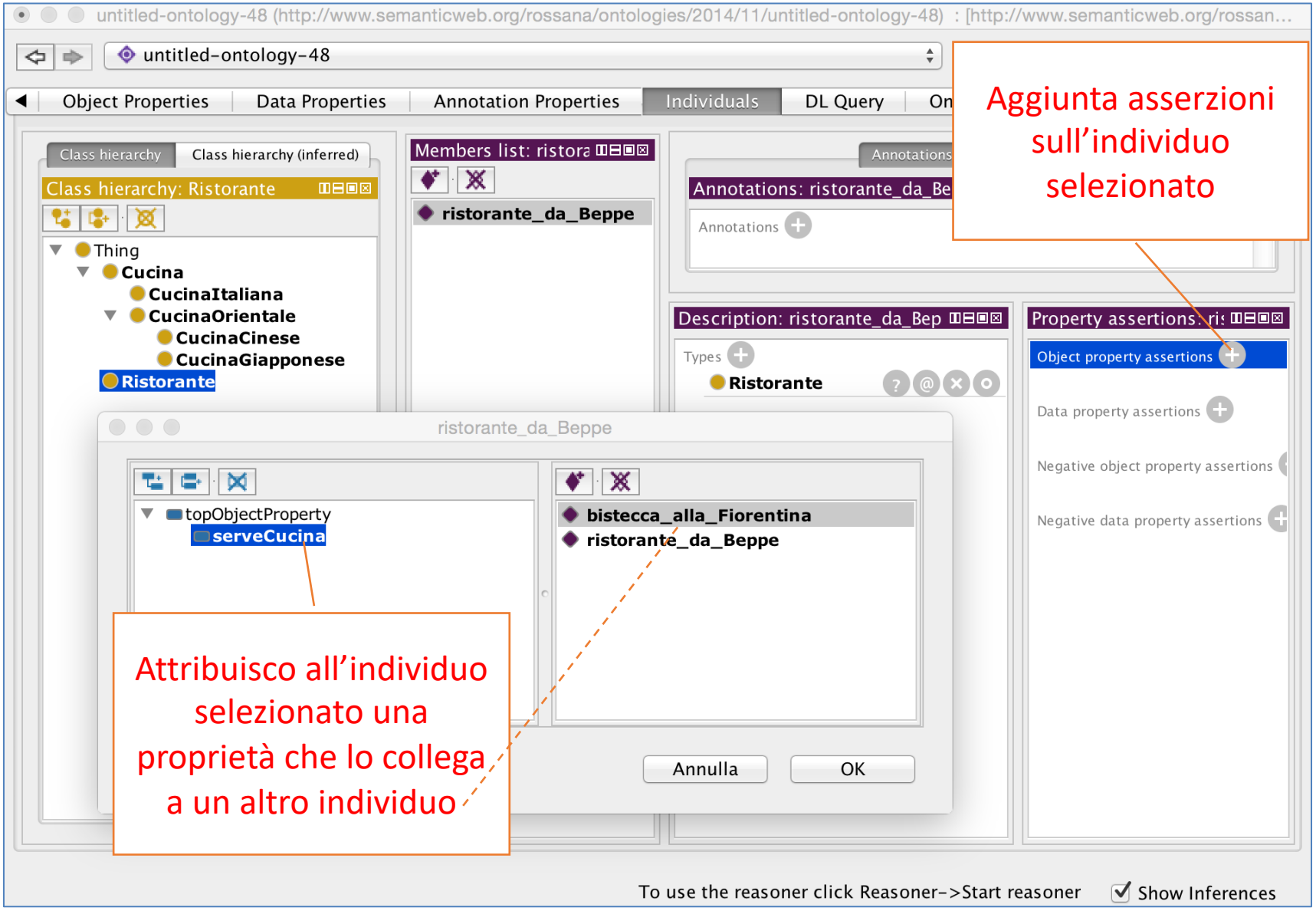


Aggiunta individui (nella classe selezionata)

Inserimento Individui

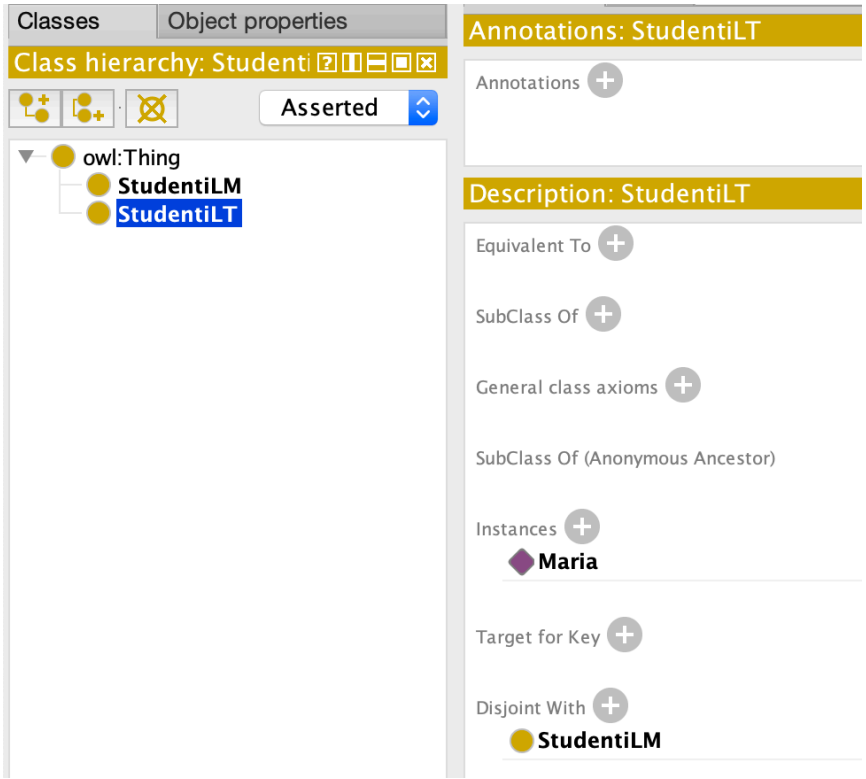


Asserzioni sugli individui



Esempio: disgiunzione

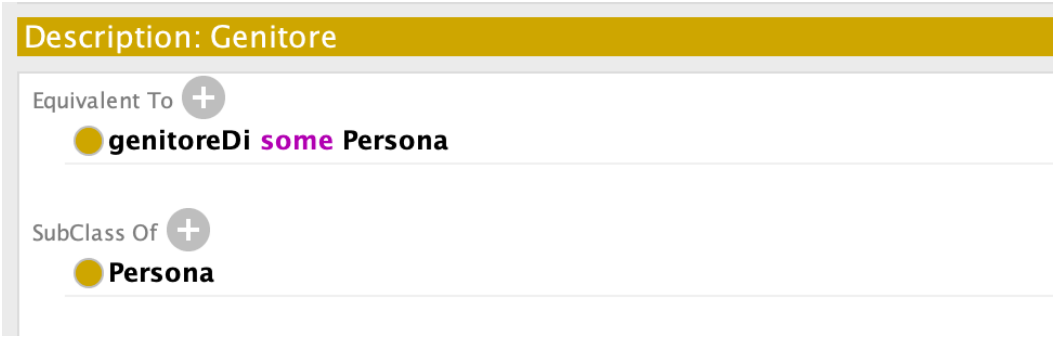
- Colloco lo stesso individuo in classi disgiunte
- Il reasoner mi segnala che l'ontologia è inconsistente



Dato che le classi StudentiLT e StudentiLM sono disgiunte, non posso collocare lo stesso individuo in entrambe

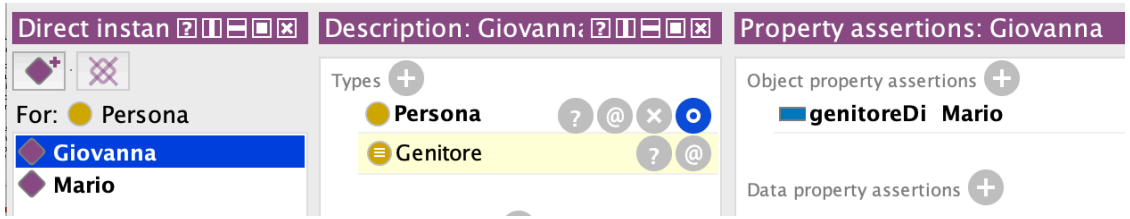
Reason for inconsistency: individual Maria is forced to belong to class "StudentiLT" and its complement

Inserimento di individui



- Inserendo un individuo con le caratteristiche di una certa classe, l'individuo viene classificato come appartenente alla classe dal *reasoner*
- Anche se è stato collocato manualmente in una classe più generale

Classificazione automatica



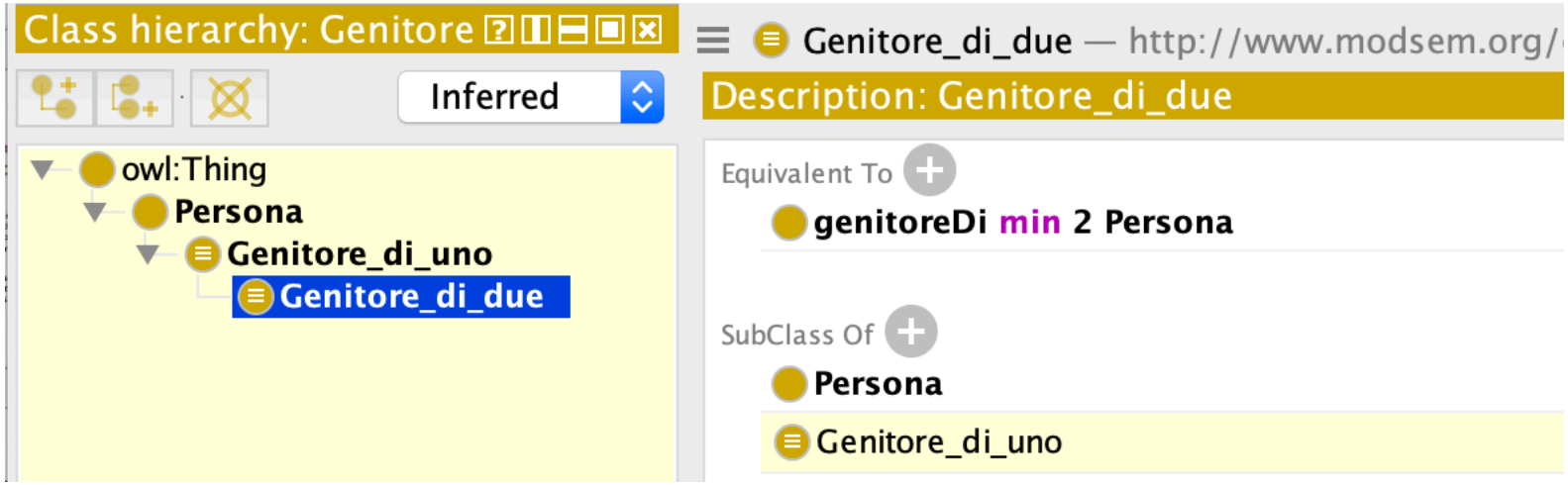
- L'individuo Giovanna è stato spostato nella classe genitore (classe più specifica di Persona in cui era stato collocato)
  - Perché Giovanna "genitoreDi" Mario

- L'inferenza scatta perché la classe Genitore è una classe *definita*, a cui sono associate condizioni necessarie e sufficienti

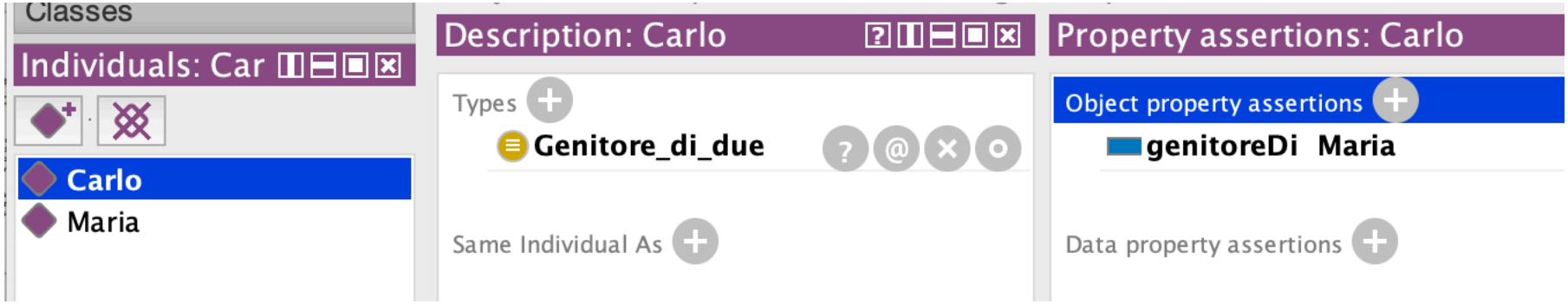
Assunzione di mondo aperto

- Le inferenze effettuate dal reasoner possono sembrare poco intuitive in alcuni casi
- La maggior parte dei sistemi di ragionamento automatico segue l'assunzione di mondo chiuso:
  - Ciò che non è rappresentato esplicitamente nel sistema viene assunto falso
  - Provate a salire su un volo se non siete nella lista dei passeggeri
- Il ragionamento sulle ontologie OWL segue invece l'assunzione di mondo aperto:
  - Il fatto che un'informazione non sia rappresentata nel sistema non determina che essa sia assunta falsa.

Individui e classi: esempio



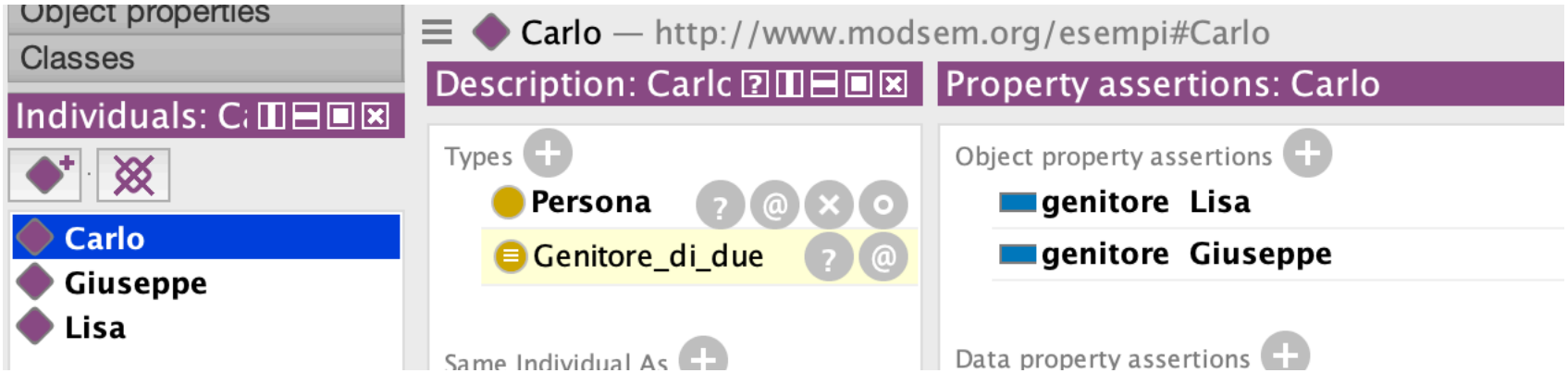
La classe Genitore_di_due è una classe definita su cui vale il vincolo che per i suoi appartenenti la relazione genitoreDi abbia almeno due individui diversi



Il fatto che un certo individuo, Carlo, sia genitoreDi un solo individuo, Maria, non genera una inconsistenza, anche se Carlo è stato collocato nella classe Genitore_di_due

Il ragionatore infatti, applica l'assunzione di mondo aperto: non viene asserito che Carlo non abbia altri figli oltre a Maria e quindi l'appartenenza alla classe, se asserita esplicitamente, non è in conflitto con le proprietà dell'individuo Carlo

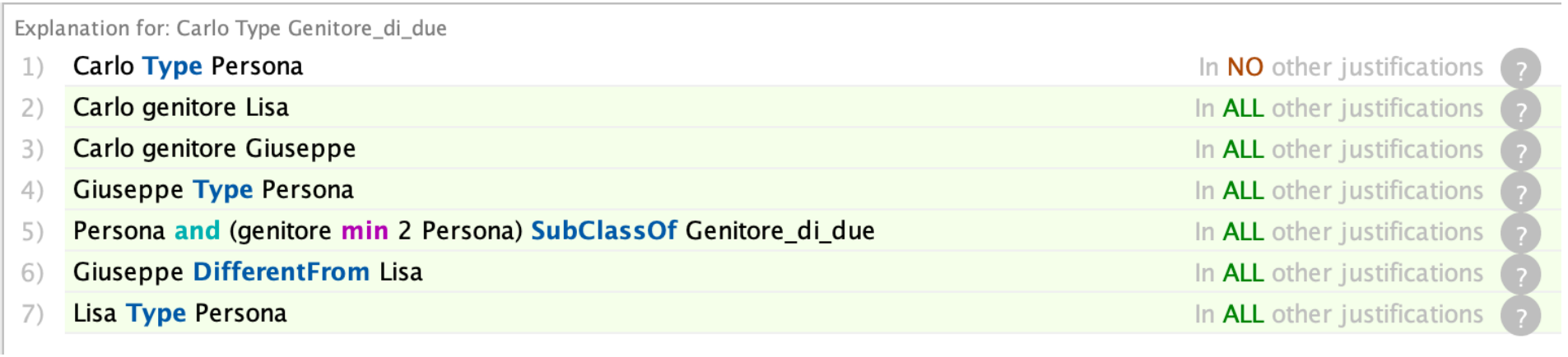
Same e different: conseguenze



Il reasoner inferisce correttamente che Giovanni, in quanto genitore di due Samuele e Carla, è appartiene alla classe Genitore_di_due
Ma solo se Samuele e Carla sono stati specificati esplicitamente come due diversi individui.

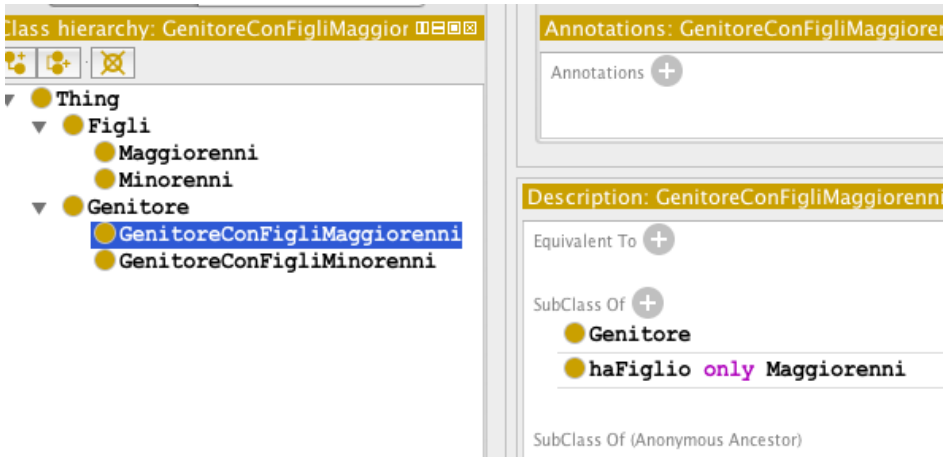
Altrimenti (mondo aperto) potrebbero essere lo stesso individuo e quindi il vincolo non sarebbe soddisfatto.

Spiegazioni (funzione Explain inferences)



- Normalmente sono presenti Explainations diverse, costruite a partire da sequenze di assiomi diverse.
- Le giustificazioni sono diverse ma in parte sovrapposte
- Utile osservare gli assiomi che fanno parte di tutte le giustificazioni.

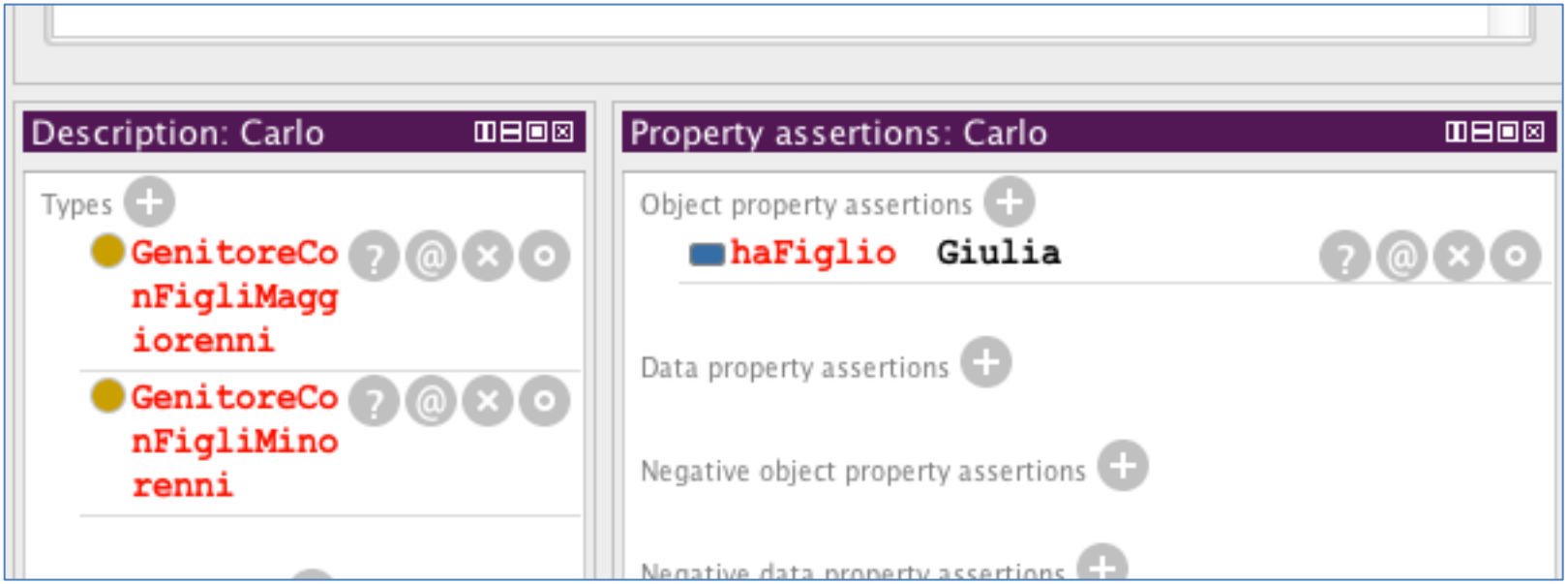
Quantificatore universale (Only)



- Le classi GenitoreConFigliMaggiorenni e GenitoreConFigliMinorenni non sono disgiunte, ma lo sono le classi Maggiorenni e Minorenni

- Se un individuo viene collocato in entrambe le classi, il reasoner rileva una inconsistenza

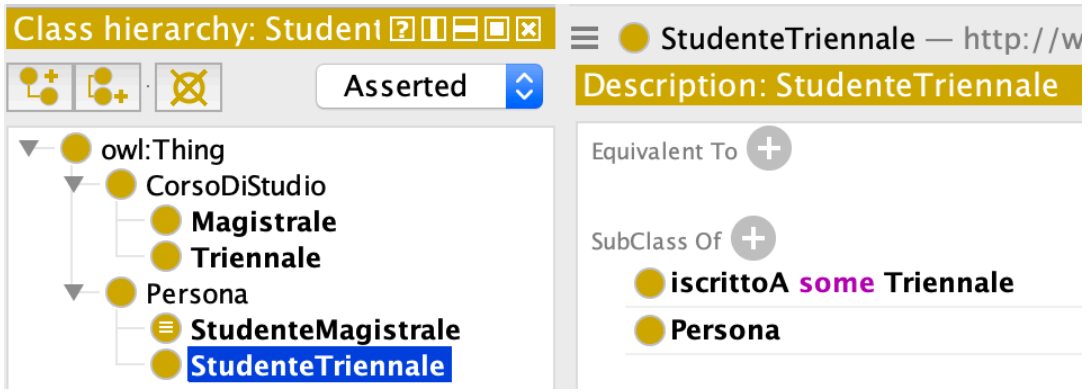
Esempio



Se Carlo, che haFiglio Giulia (minorenne), viene dichiarato come membro di entrambe le classi, il reasoner rileva un'inconsistenza.

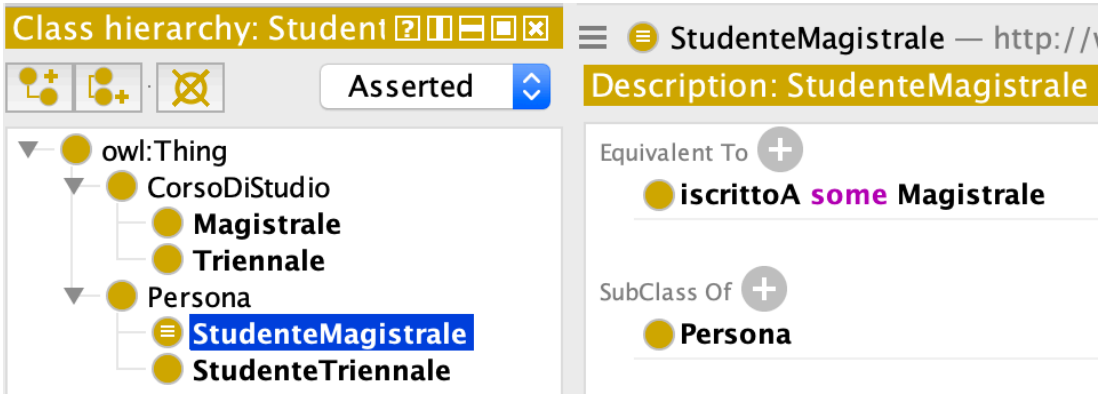
Giulia infatti si troverebbe a appartenere a due classi distinte, che sono state dichiarate come disgiunte.

Classe primitiva



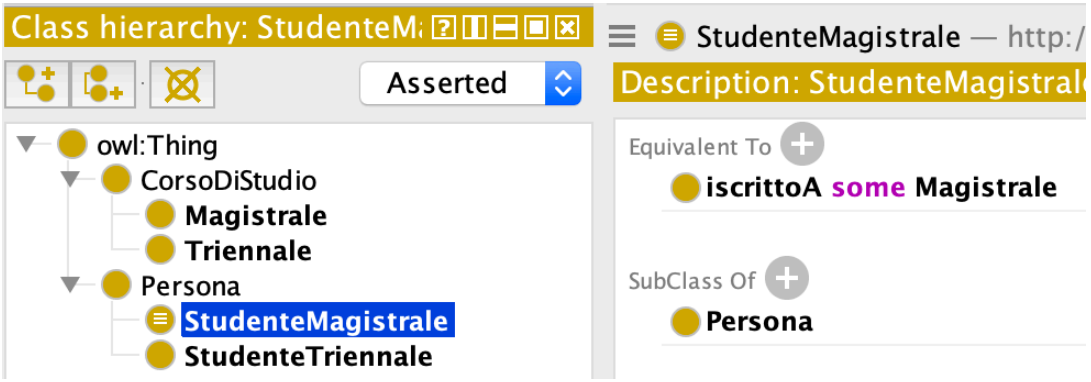
La classe primitiva può avere condizioni necessarie associate ad essa (qui, iscrittoA, vincolato al quantificatore esistenziale), dichiarate tramite lo slot SubClass Of.

Classe definita



La classe **definita** è una classe equivalente a un insieme di condizioni necessarie e sufficienti, specificate mediante **Equivalent To**

Classi primitive e definite



- Le classi primitive (StudenteTriennale nell'esempio) hanno solo condizioni necessarie associate
- Le classi definite (Studente Magistrale) hanno condizioni necessarie e sufficienti associate → permettono al reasoner di inferire l'appartenenza alla classe di individui che hanno le caratteristiche elencate come condizioni necessarie e sufficienti della classe.

Esempio

- Se un individuo che ha anche figli non adolescenti (e questi non possono essere assunti tali!) viene descritto come “genitore di figli solo adolescenti” il reasoner rileva un’inconsistenza

Attenzione a quell'only!

- Only **non** significa almeno uno
- (Dal Primer:) "There is one particular **misconception** concerning the universal role restriction. As an example, consider the happiness axiom:" (una persona felice è una persona che ha solo figli felici):

  ObjectAllValuesFrom(:hasChild :HappyPerson)

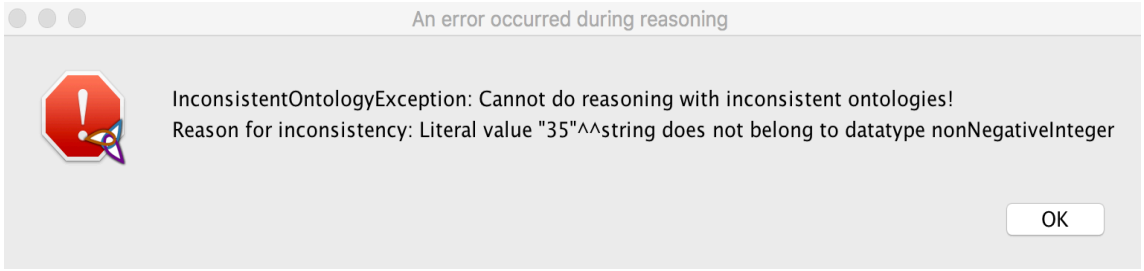
- "The intuitive reading suggests that in order to be happy, a person must have *at least* one happy child. Yet, this is not the case: any individual that is not a "starting point" of the property hasChild is a class member of any class defined by universal quantification over hasChild."

Ragionamento su data properties

- I valori assegnati a un individuo per una certa data property devono appartenere al data type specificato come range
- Se il datatype è diverso da quello previsto, il reasoner rileva un'inconsistenza

Esempio

- La proprietà *età* ha come range xsd:nonNegativeInteger
- Se la proprietà ha un valore appartenente a un datatype diverso (es. xsd:string), l'ontologia è inconsistente



Restrizioni su data properties

- E’ possibile porre una restrizione su una data property
  - Usando quantificatori e cardinalità tuttavia si esprime un vincolo sul numero delle relazioni, non sul valore che assume la proprietà
- Per creare classi definite in molti casi si usano i pattern che individuano un range specifico di valori rispetto al tipo di dato
  - Per esempio le stringhe che cominciano con una certa lettera, gli interi superiori a un certo valore, ecc.
  - Valgono le specifiche su XSD Datatypes
  - E’ possibile creare specifici datatypes (es. un range di valori)

Restrizioni su data properties: esempio

```
owl:onProperty :età ;
owl:someValuesFrom [ rdf:type rdfs:Datatype ;
                     owl:onDatatype xsd:nonNegativeInteger ;
                       owl:withRestrictions ( [ xsd:minInclusive "18"^^xsd:nonNegativeInteger
                                              ]
                                            )
                     ]
```

Property chain

In OWL2 è possibile creare delle catene di proprietà

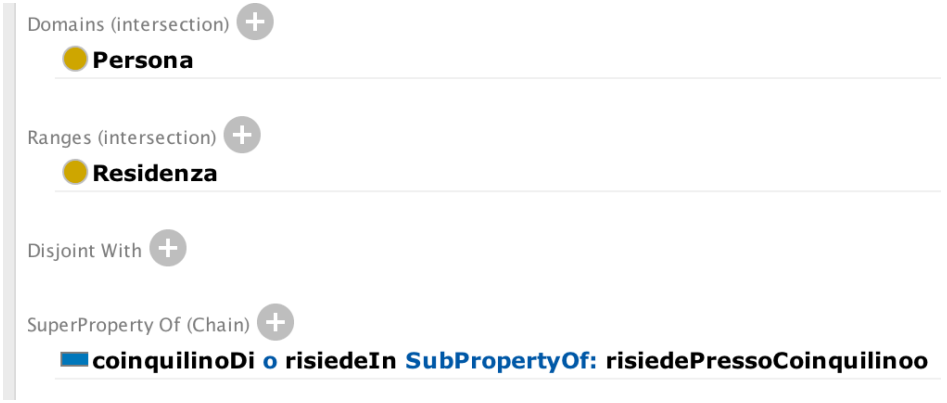
:hasGrandparent owl:propertyChainAxiom (:hasParent :hasParent) .

Se si asserisce che

- *a* hasParent *b*
- *b* hasParent *c*

Un reasoner inferisce che *a* hasGrandParent *c*

Esempio property chain



- Se una persona A è coinquilina di una persona B e la persona B risiede in un certo luogo, anche la persona A risiede in quel luogo

Classi primitive e ragionamento

LE CLASSI PRIMITIVE NON PERMETTONO AI REASONER DI COLLOCARE GLI INDIVIDUI NELLE CLASSI IN MODO AUTOMATICO

TUTTAVIA, ESSE PERMETTONO AI REASONER DI INDIVIDUARE INCONSISTENZE CON LE ASSERZIONI SUGLI INDIVIDUI

Esempio di modellazione (1)

- Esistono **città** e **nazioni**. Le nazioni possono essere **nazioni europee**, **asiatiche**, **africane**, ecc.
- Le città sono *situate* nelle nazioni. Hanno un *nome* e una *data di fondazione*.
- Le città *hanno* una **popolazione**. La popolazione può essere una **popolazione ampia**, **media** o **scarsa**
- Le **città grandi** sono quelle che hanno una popolazione grande, quelle **medie** sono le città che hanno una popolazione media, ecc.
- Le **città europee** sono quelle situate in una nazione europea, quelle asiatiche sono le città situate in una nazione asiatica, ecc.

Esempio di modellazione (2)

- Torino è situata in Italia e ha una popolazione media
- Tokio è situata in Giappone e ha una popolazione ampia
- Bergamo è una città piccola
- Suggerimenti: codificare nome e data di fondazione come data properties