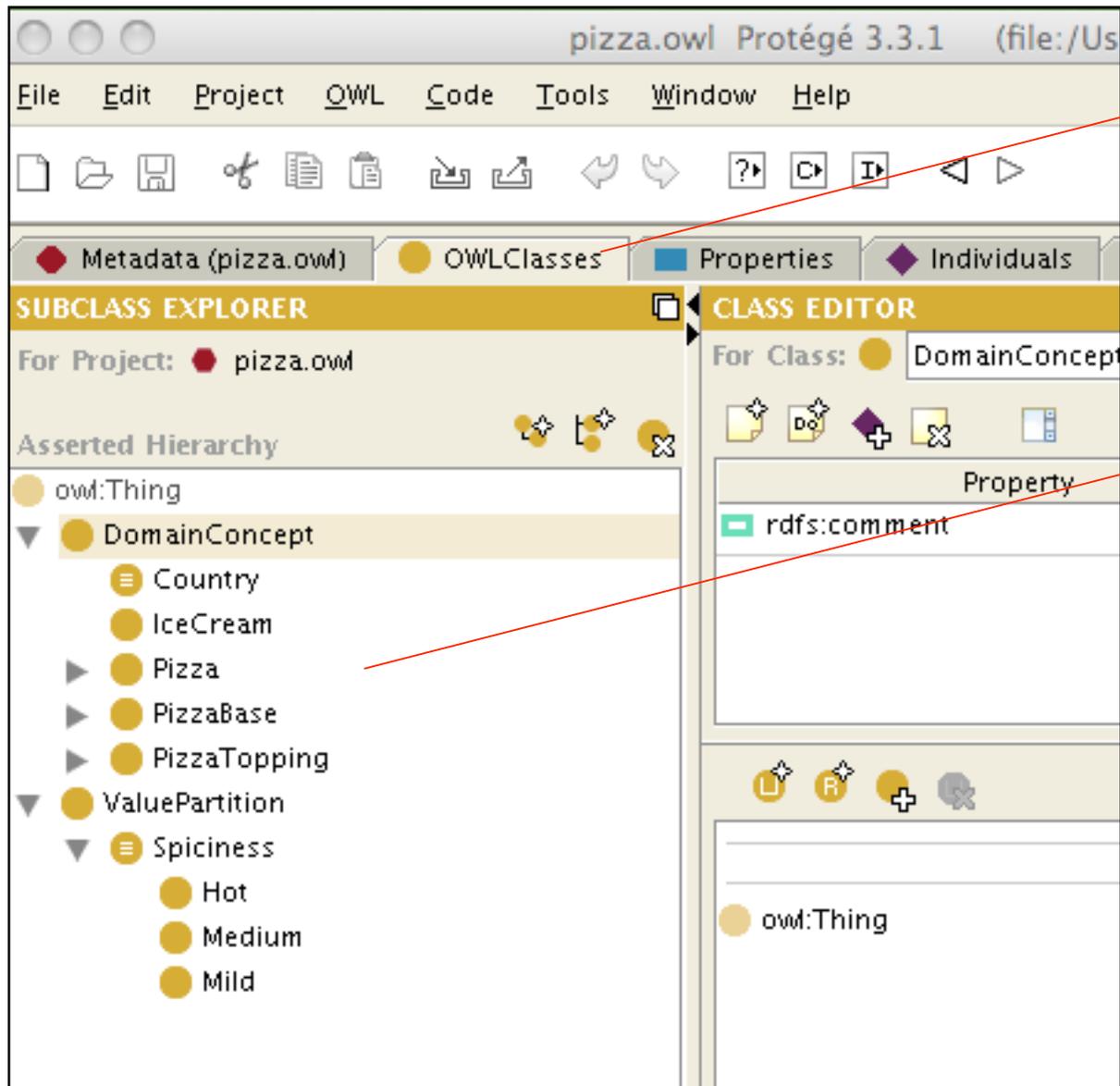


Protégé

Editor per la creazione di
ontologie



Tab separati per
Classi
Proprietà
Individui

Es. struttura
gerarchica delle
classi (solo
livello sottostante
a "Domain
Concept"

The screenshot displays an ontology editor interface. At the top, there are tabs for 'Metadata (pizza.owl)', 'OWLClasses', 'Properties', 'Individuals', and 'Forms'. The 'SUBCLASS EXPLORER' on the left shows a tree view of classes under the project 'pizza.owl'. The 'CLASS EDITOR' on the right is currently viewing the class 'FruttiDiMare'. It shows a table of properties for this class, including 'rdfs:comment' and 'rdfs:label' (with the value 'FrutosDoMar'). Below the properties, there is a list of subclasses and restrictions for 'FruttiDiMare', including 'NamedPizza' and several 'hasTopping' restrictions with 'only', 'some', and 'or' operators. Two red arrows point from the text below to the 'FruttiDiMare' class in the hierarchy and the 'hasTopping' restrictions in the class editor.

Classi specifiche di Pizze (sottoclassi di “NamedPizza”, cioè ‘pizze con nome’): American, American Hot

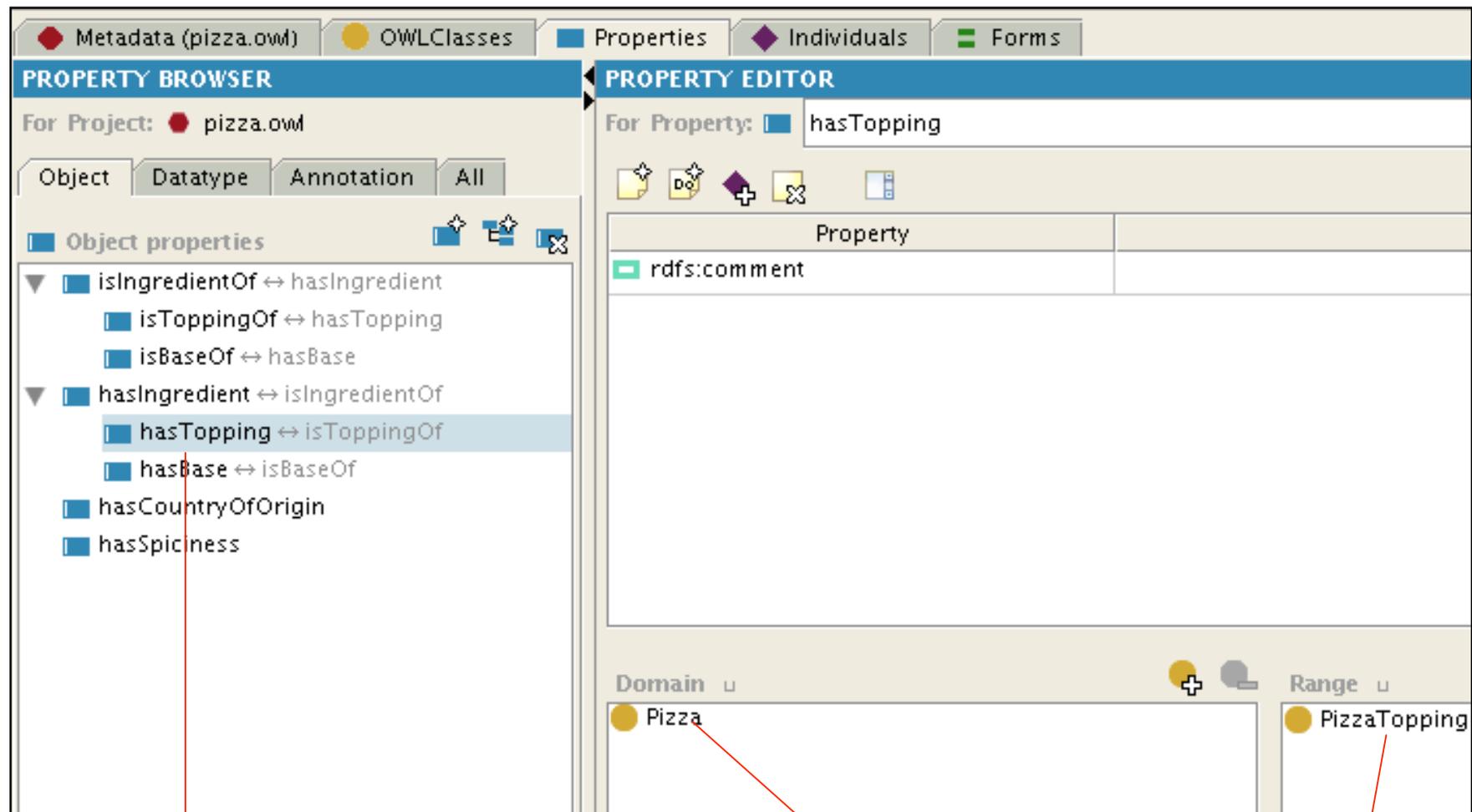
La classe FruttiDiMare ha come farcitura Tomato Topping, MixedSeafoodTopping e GarlicTopping e una generica base

The screenshot shows a software interface with a list of pizza types on the left and a detailed view of the 'Capricciosa' pizza class on the right. The right panel displays 'Asserted Conditions' for 'NamedPizza' and 'hasBase some PizzaBase' (inherited from Pizza).

Property	Value	Category
NamedPizza		NECESSARY
hasTopping	only (TomatoTopping or HamTopping or MozzarellaTopping or AnchoviesTopping or OliveTopping)	NECESSARY
hasTopping	some PeperonataTopping	NECESSARY
hasTopping	some TomatoTopping	NECESSARY
hasTopping	some HamTopping	NECESSARY
hasTopping	some AnchoviesTopping	NECESSARY
hasTopping	some CaperTopping	NECESSARY
hasTopping	some MozzarellaTopping	NECESSARY
hasTopping	some OliveTopping	NECESSARY
hasBase	some PizzaBase	INHERITED [from Pizza]

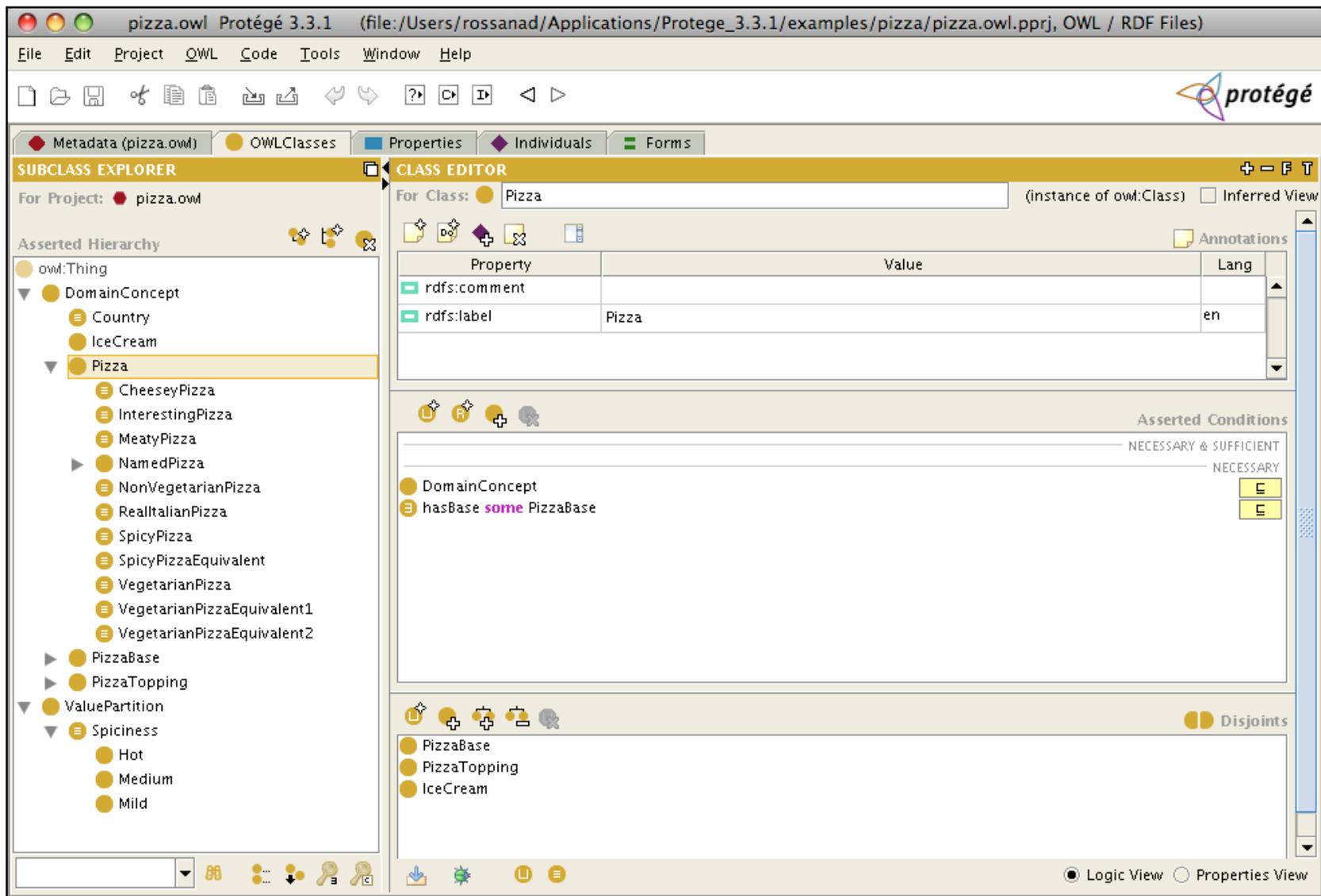
Alcune proprietà di una classe sono specifiche di quella classe, altre sono ereditate
 Per esempio la pizza Capricciosa eredita dalla classe Pizza la proprietà di avere una base di qualche tipo (some PizzaBase)

(Alcune pizze invece hanno tipi specifici di basi, cioè basi sottili o spesse)



Le proprietà mettono le classi in relazione tra di loro, per esempio i tipi di pizza con gli ingredienti.

Anche le relazioni hanno una struttura gerarchica. Per esempio, la relazione `hasTopping` è un sottotipo di `hasIngredient` che serve per identificare gli ingredienti della farcitura. `hasIngredient` mette in relazione pizze (Domain:pizza) con farciture (range:PizzaTopping). Il suo inverso è `isIngredientOf`.



PizzaBase, PizzaTopping e IceCream sono classi disgiunte, mentre i tipi di pizza (InterestingPizza, MeatyPizza, ecc.) non lo sono (una pizza può essere 'interessante' ma anche 'a base di carne', meaty)

MeatyPizza
NamedPizza
NonVegetarianPizza
RealltalianPizza
SpicyPizza
SpicyPizzaEquivalent
VegetarianPizza
VegetarianPizzaEquivalent1
VegetarianPizzaEquivalent2
PizzaBase
PizzaTopping
Spiciness
Hot

Pizza

- not (hasTopping some FishTopping)
- not (hasTopping some MeatTopping)
- hasBase some PizzaBase

NECESSARY & SUFFICIENT
NECESSARY
INHERITED [from Pizza]

Disjoints

... ma una pizza vegetariana (che non ha una farcitura di pesce né di carne) ovviamente è una classe disgiunta da quella delle pizze non vegetariane

IceCream
Pizza
CheeseyPizza
InterestingPizza
MeatyPizza
NamedPizza
NonVegetarianPizza
RealltalianPizza
SpicyPizza
SpicyPizzaEquivalent
VegetarianPizza

Pizza

- hasTopping min 3
- hasBase some PizzaBase

least 3
Pizzalt

Una pizza interessante è una pizza che ha almeno 3 ingredienti