

# Reasoning with OWL

Olivier Dameron<sup>1</sup>, Alan Rector<sup>2</sup>, Nick Drummond<sup>2</sup>,  
Matthew Horridge<sup>2</sup>



- 1) EA 3888, Université de Rennes1, France
- 2) The University of Manchester, United Kingdom

## Credits:

- Matthew Horridge, Holger Knublauch et al.  
*A Practical guide to building OWL ontologies using  
the Protégé-OWL plugin and CO-ODE tools*
- Natasha Noy, Alan Rector  
*W3C “Semantic Web Best Practice” Working Group*

# Objective

**Acquiring an in-depth understanding of the OWL-DL semantics in order to perform advanced reasoning tasks**

- We will rely on the pizza example for:
  - a better formalization of the domain knowledge
  - leveraging OWL-DL reasoning capabilities for an easier curation of the ontology
  - an overview of some good practice

# Outline

- OWL semantics
- Open world assumption
- Reasoning with individuals

# Getting started

## 1. Getting Protégé

- version 4.0
- <http://protege.stanford.edu>

## 2. Getting some documentation

- <http://protege.stanford.edu/doc/users.html>
- OWL Tutorial : <http://www.co-ode.org>
- Wiki: <http://protege.cim3.net/cgi-bin/wiki.pl>
- Mailing lists

# Getting started

1. Use the protege2007owlTutorial-01.owl ontology from:  
<http://www.ea3888.univ-rennes1.fr/dameron/protege2007/>
2. Launch Protégé
3. Select “Open OWL ontology”
4. Retrieve your local copy of the ontology

# OWL Semantics

*(the theoretical part)*

# Individuals

- Atoms
- Individuals have an identity and can be counted
- They are fundamental for understanding the semantics of DL...
- ... but you hardly use them when building ontologies

# Properties

- A property = binary relationships btw individuals
- Domain, Range
  - Used as axioms (e.g. *hasTopping* and ice creams)
- Subproperties
- Characteristics
  - Transitive: e.g. *hasPart*, *hasAncestor...*
  - Symmetric: e.g. *isSiblingOf...*
  - Functional: e.g. *hasSSN*, *hasMother...*

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-01.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Object Properties: hasIngredient

- + hasIngredient
  - hasBase
  - hasTopping
- isIngredientOf
  - isBaseOf
  - isToppingOf

Annotations: hasIngredient

Annotations

comment

NB Transitive – the ingredients of ingredients are ingredients of the whole@en

Annotations Object Property Usage

Character

- Functional
- Inverse functi...
- Transitive
- Symmetric
- Antisymmetric
- Reflexive
- Irreflexive

Description: hasIngredient

Equivalent object properties +

Super properties +

Inverse properties +

**isIngredientOf** x o

Disjoint properties +

Property chains +

Domains and ranges: hasIngredient

Domains (intersection) +

**Food** x o

Ranges (intersection) +

**Food** x o

The screenshot shows the Protege 2007 OWL tutorial interface. The main window displays the 'hasIngredient' object property, which is marked as Transitive. Annotations provide a note about transitivity and its English description. The 'Character' section shows various property types, with 'Transitive' checked. The 'Description' section lists equivalent properties, super properties, inverse properties, disjoint properties, and property chains. The 'Domains and ranges' section shows that both the domain and range are set to the 'Food' class.

# Functional Properties

- Functional property: each element of the domain can have 0 or 1 image in the range
  - ex: *hasBiologicalMother, isToppingOf, isBaseOf,...*
- If a property is functional, then its inverse is inverse functional
  - ex: *hasTopping*
- A property can be both functional and inv.-functional
  - ex: *hasSSN, hasBase*
  - not all do! -> *hasBiologicalMother*

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-01.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Object Properties: isToppingOf

- hasIngredient
  - hasBase
  - hasTopping
- isIngredientOf
  - isBaseOf
  - isToppingOf

Annotations: isToppingOf

Annotations

comment

Any given instance of topping should only be added to a single pizza (no cheap half-measures on our pizzas)@en

Annotations Object Property Usage

Character

Functional

Inverse functi...

Transitive

Symmetric

Antisymmetric

Reflexive

Irreflexive

Description: isToppingOf

Equivalent object properties

Super properties

- isIngredientOf

Inverse properties

- hasTopping

Disjoint properties

Property chains

Domains and ranges: isTopping

Domains (intersection)

- PizzaTopping

Ranges (intersection)

- Pizza

Character

Functional

Inverse functi...

Transitive

Symmetric

Antisymmetric

Reflexive

Irreflexive

Description: isToppingOf

Equivalent object properties

Super properties

- isIngredientOf

Inverse properties

- hasTopping

Disjoint properties

Property chains

Domains and ranges: isTopping

Domains (intersection)

- PizzaTopping

Ranges (intersection)

- Pizza

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-01.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Object Properties: hasTopping

- hasIngredient
  - hasBase
  - hasTopping
- isIngredientOf
  - isBaseOf
  - isToppingOf

Annotations: hasTopping

Annotations

comment

Note that hasTopping is inverse functional because isToppingOf is functional@en

Annotations Object Property Usage

Character

- Functional
- Inverse functi...
- Transitive
- Symmetric
- Antisymmetric
- Reflexive
- Irreflexive

Description: hasTopping

Equivalent object properties +

Super properties +

- hasIngredient

Inverse properties +

- isToppingOf

Disjoint properties +

Property chains +

Domains and ranges: hasTopping

Domains (intersection) +

- Pizza

Ranges (intersection) +

- PizzaTopping

Character

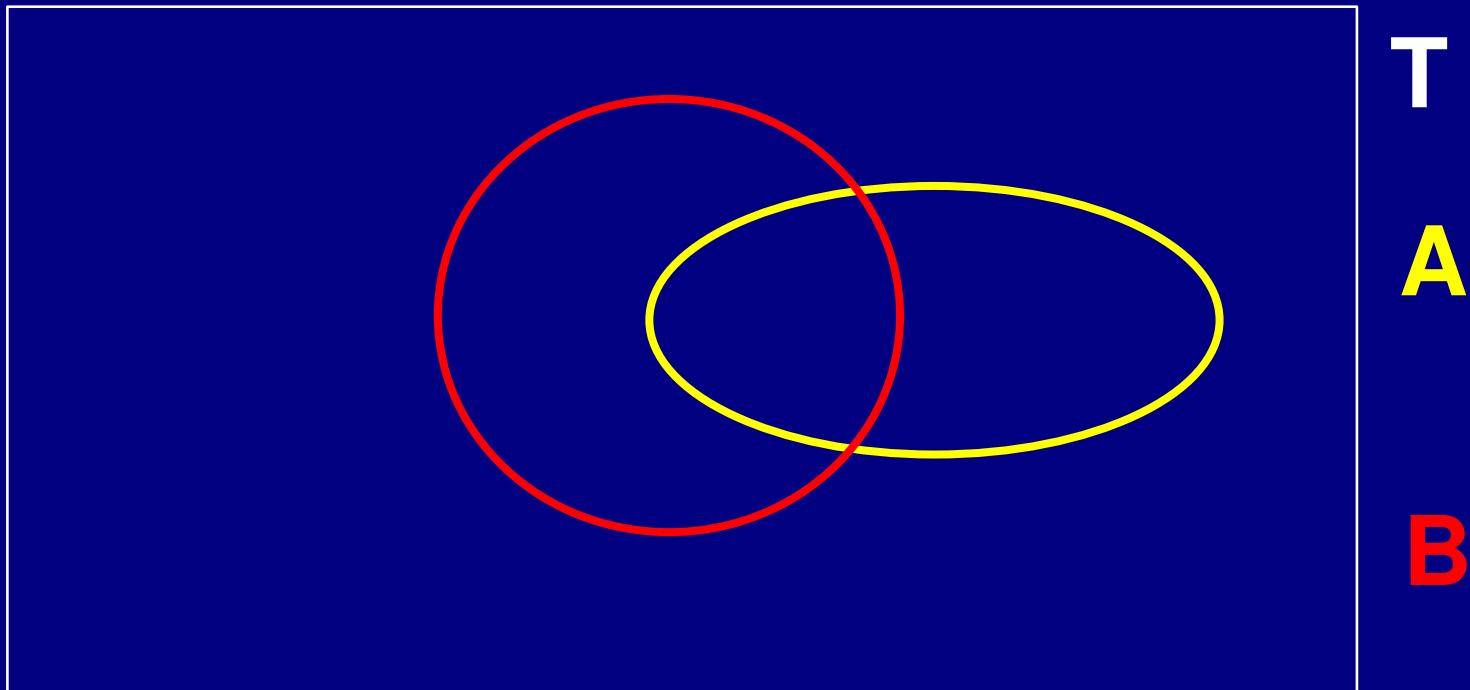
Description: hasTopping

Domains and ranges: hasTopping

# Classes

- A class is a set of individuals
  - Special classes:
    - top ( $\top$ ) = owl:Thing i.e. set of all the individuals
    - bottom ( $\perp$ ) = empty set
  - Can be combined using set operators
    - subset (subsumption)
    - disjoint sets
    - union
    - intersection
    - complement

# Classes: Disjointness



By default, any individual MAY be an instance of any classes => partial overlap of classes is assumed

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-01.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: CheeseTopping

- Thing
- DomainConcept
- Food
  - Pizza
  - PizzaBase
  - PizzaTopping
    - CheeseTopping
    - FishTopping
    - FruitTopping
    - HerbSpiceTopping
    - MeatTopping
    - NutTopping
    - SauceTopping
    - VegetableTopping
    - VegetarianTopping

Class Annotations: CheeseTopping

Annotations +

Class Annotations Class Usage

Class Description: CheeseTopping

Superclasses +

- PizzaTopping

Inherited anonymous classes

Instances +

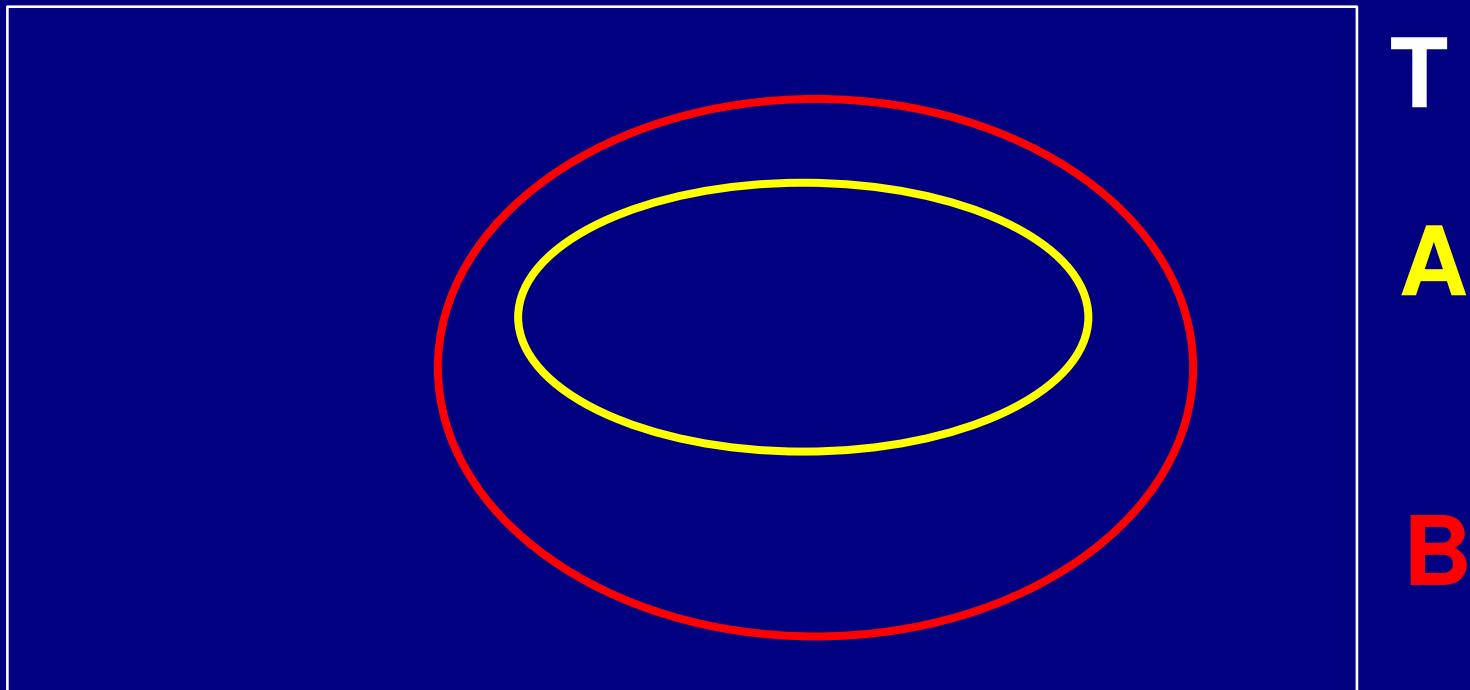
Disjoint classes +

- FruitTopping
- SauceTopping
- VegetableTopping
- MeatTopping
- NutTopping
- FishTopping
- HerbSpiceTopping

Asserted class hierarchy Inferred class hierarchy

The screenshot displays the Protege 2007 OWL tutorial interface. The top navigation bar includes File, Edit, Reasoner, Tools, Refactor, Tabs, View, Window, and Help. Below the tabs are Active Ontology, Entities, Classes, Object Properties, Data Properties, Individuals, OWLViz, and DL Query. The main area shows the asserted class hierarchy for 'CheeseTopping', which includes Thing, DomainConcept, Food (with Pizza, PizzaBase, PizzaTopping), and CheeseTopping itself. Below CheeseTopping are FishTopping, FruitTopping, HerbSpiceTopping, MeatTopping, NutTopping, SauceTopping, VegetableTopping, and VegetarianTopping. A red circle highlights the 'Disjoint classes' section in the middle-right panel, which lists FruitTopping, SauceTopping, VegetableTopping, MeatTopping, NutTopping, FishTopping, and HerbSpiceTopping.

# Classes: subsumption



$A \subset B$  : all the instances of A are instances of B (A is subClass of B)

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-01.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: GreenPepperTopping

Annotations

Class Annotations: GreenPepperTopping

Class Description: GreenPepperTopping

Equivalent classes

Superclasses

PepperTopping

Inherited anonymous classes

Instances

Disjoint classes

SweetPepperTopping

PeperonataTopping

JalapenoPepperTopping

Asserted class hierarchy Inferred class hierarchy

The screenshot displays the Protege 2007 interface for editing an ontology. The left pane shows the asserted class hierarchy, which is a tree structure of food-related classes. The right pane is divided into two main sections: 'Class Annotations' for the selected class 'GreenPepperTopping' and 'Class Description' for the same class.

**Asserted Class Hierarchy:**

- Food
  - Pizza
    - PizzaBase
  - PizzaTopping
    - CheeseTopping
    - FishTopping
    - FruitTopping
    - HerbSpiceTopping
    - MeatTopping
    - NutTopping
    - SauceTopping
    - VegetableTopping
      - ArtichokeTopping
      - AsparagusTopping
      - CaperTopping
      - GarlicTopping
      - LeekTopping
      - MeatyVegetableTopping
      - MushroomTopping
      - OliveTopping
      - OnionTopping
      - PepperTopping
        - GreenPepperTopping
        - HotGreenPepperTop
        - JalapenoPepperTopping
        - PeperonataTopping
        - SweetPepperTopping

# Classes

- Cumulative approach: combine classes
  - using set operators (union, intersection, complement)
  - express constraints
  - define complex concepts
- Intensional approach: describe the characteristics of a class and the system will automatically:
  - recognize that an individual is an instance of it
  - recognize that it is a subclass or a superclass of another class

# Combining Classes

# Objective

- Combine classes using the OR, AND and NOT operators
- Refer to the semantics of these operators (and avoid some basic mistakes)

=> find out which pizza are:

- Cheesy and vegetarian
- Cheesy or vegetarian
- vegetarian and not vegetarian

# Prerequisite

- Pizza
  - | • VegetarianPizza NOT DISJOINTS
  - | • CheesyPizza
- NamedPizza
  - | • MargheritaPizza DISJOINTS
  - | • AmericanPizza
  - | • CaprinaPizza

Don't worry about the toppings, this is the next step!

# AND (Intersection)

- Create `CheesyAndVegetarianPizza` as a subclass of `Pizza`
  - so far, except for the name, we have not provided any meaning
  - we have not exploited the cumulative approach
- Add the necessary condition:  
 $\text{VegetarianPizza} \sqcap \text{CheesyPizza}$
- Classify

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-01.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data

Asserted Class Hierarchy: CheesyAndVegetarianPizza

Superclasses +

Pizza

VegetarianPizza  
and CheeseyPizza

Class Description: CheesyAndVegetarianPizza

Equivalent classes +

Superclasses +

Pizza

VegetarianPizza  
and CheeseyPizza

Inherited anonymous classes

hasBase some PizzaBase

Pizza  
that hasTopping only VegetarianTopping

Pizza  
that hasTopping some CheeseTopping

Instances +

Disjoint classes +

Thing

DomainConcept

Food

Pizza

CheesyAndVegetarianPizza

CheeseyPizza

MeatyPizza

NamedPizza

ThinAndCrispyPizza

VegetarianPizza

PizzaBase

PizzaTopping

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties

Inferred class hierarchy: CheesyAndVegetarianPizz

PIZZA

CheeseyPizza

CheesyAndVegetarianPizza

MeatyPizza

NamedPizza

ThinAndCrispyPizza

VegetarianPizza

CheesyAndVegetarianPizza

Thing

Nothing

DomainConcept

Food

Pizza

CheeseyPizza

CheesyAndVegetarianPizza

MeatyPizza

NamedPizza

ThinAndCrispyPizza

VegetarianPizza

CheesyAndVegetarianPizza

PizzaBase

PizzaTopping

PIZZA

CheeseyPizza

CheesyAndVegetarianPizza

MeatyPizza

NamedPizza

ThinAndCrispyPizza

VegetarianPizza

CheesyAndVegetarianPizza

Pizza

VegetarianPizza  
and CheeseyPizza

CheeseyPizza

VegetarianPizza

Inherited anonymous classes

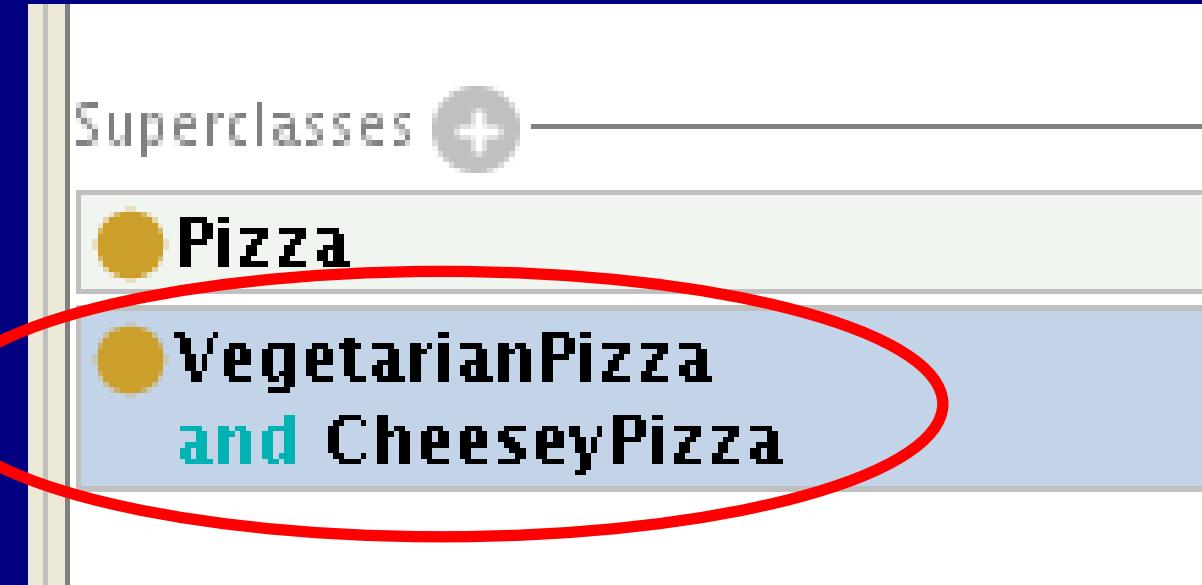
hasBase some PizzaBase

Pizza  
that hasTopping only VegetarianTopping

Instances

Disjoint classes

```
graph TD; Thing --> Nothing; Thing --> DomainConcept; DomainConcept --> Food; Food --> Pizza; Pizza --> CheeseyPizza; Pizza --> CheesyAndVegetarianPizza; Pizza --> MeatyPizza; Pizza --> NamedPizza; Pizza --> ThinAndCrispyPizza; Pizza --> VegetarianPizza; Pizza --> CheesyAndVegetarianPizza; Pizza --> PizzaBase; Pizza --> PizzaTopping;
```

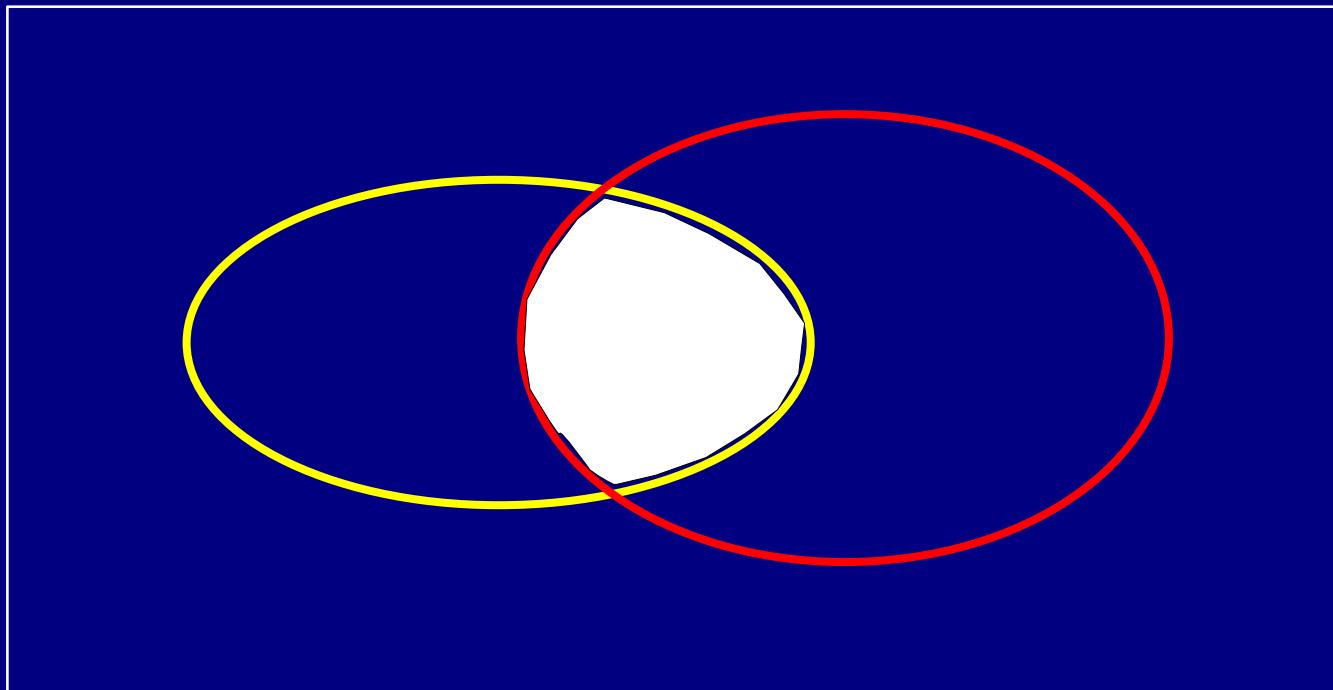


is equivalent to:



... but the reasoning would have been trivial :-)

# AND (Intersection)



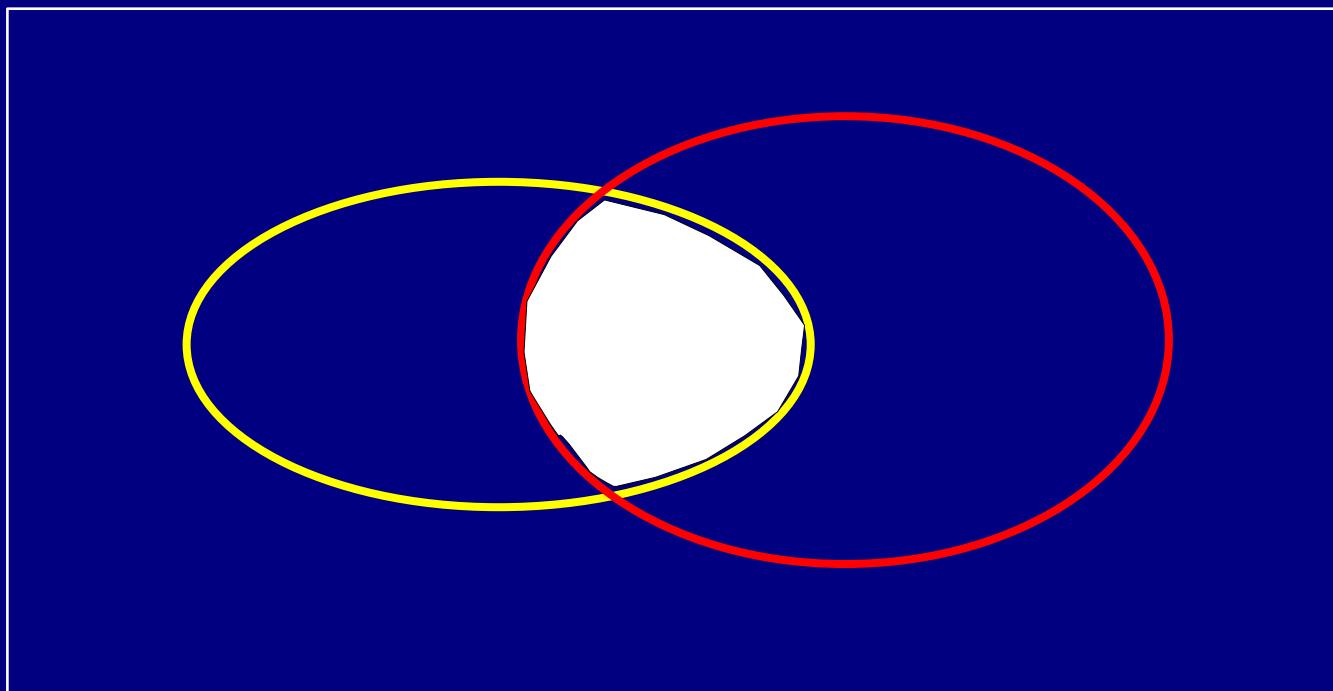
A

A  $\cap$  B

B

$A \cap B = \text{set of indiv. instances of } A \text{ and of } B$

# AND (Intersection)



A

$A \cap B$

B

Ex: VegetarianPizza  $\sqcap$  CheesyPizza

# OR (Union)

- Create CheesyOrVegetarianPizza as a subclass of Pizza
- Add the necessary condition:  
VegetarianPizza  $\sqcup$  CheesyPizza
- Classify :-()

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties

Asserted Class Hierarchy: CheeseyOrVegetarianPiz

Class Annotations: C

Annotations +

Class Annotations Class Usage

Class Description: CheeseyOrVegetarianPizza

Equivalent classes +

Superclasses +

Pizza

VegetarianPizza or CheeseyPizza

Inherited anonymous classes

hasBase some PizzaBase

Instances +

Disjoint classes +

Superclasses +

Pizza

VegetarianPizza or CheeseyPizza

Class Annotations Class Usage

Class Description: CheeseyOrVegetarianPizza

Equivalent classes +

Superclasses +

Pizza

VegetarianPizza or CheeseyPizza

Inherited anonymous classes

hasBase some PizzaBase

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

The screenshot shows the Co-ODE Ontology Editor interface. On the left, the 'Asserted Class Hierarchy' panel displays a tree structure of classes under 'CheeseyOrVegetarianPiz'. The 'Food' class has several subclasses: Pizza, CheeseyOrVegetarianPizza, CheesyAndVegetarianPizza, CheeseyPizza, MeatyPizza, NamedPizza, ThinAndCrispyPizza, VegetarianPizza, PizzaBase, and PizzaTopping. The 'CheeseyOrVegetarianPizza' class is selected. On the right, the 'Class Description' panel for 'CheeseyOrVegetarianPizza' shows its annotations, usage, and description. The 'Superclasses' section lists 'Pizza' and 'VegetarianPizza or CheeseyPizza'. A red arrow points from the 'Superclasses' section in the class description to the 'Superclasses' section in the asserted class hierarchy. Both sections show the same list of superclasses.

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Inferred class hierarchy: CheeseyOrVegetarianPizz:

Annotations +

Class Annotations Class Usage

Class Description: CheeseyOrVegetarianPizza

Equivalent classes +

Superclasses +

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

Thing

Nothing

DomainConcept

Food

Pizza

CheeseyOrVegetarianPizza

CheeseyPizza

MeatyPizza

NamedPizza

ThinAndCrispyPizza

VegetarianPizza

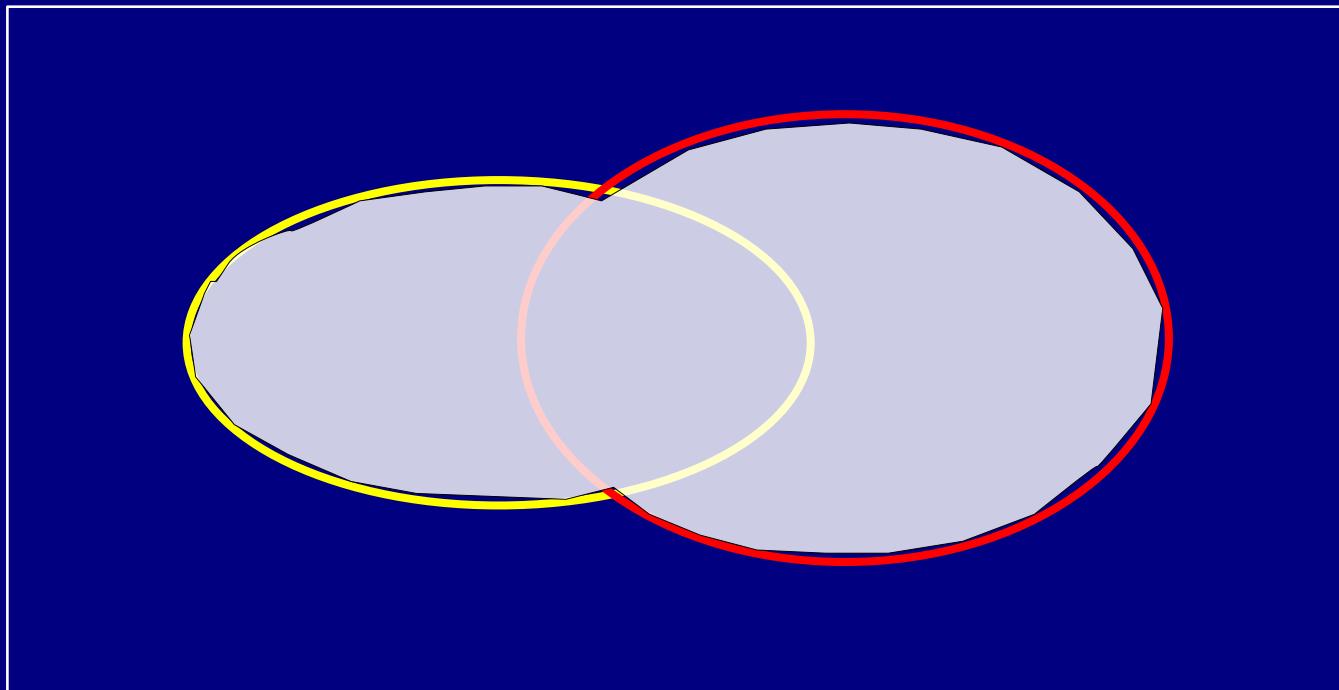
PizzaBase

PizzaTopping

A red circle highlights the class **CheeseyOrVegetarianPizza** in the inferred class hierarchy tree.

**CheesyPizza and VegetarianPizza are not recognized as subclasses of CheeseyOrVegetarianPizza**

# OR (Union)



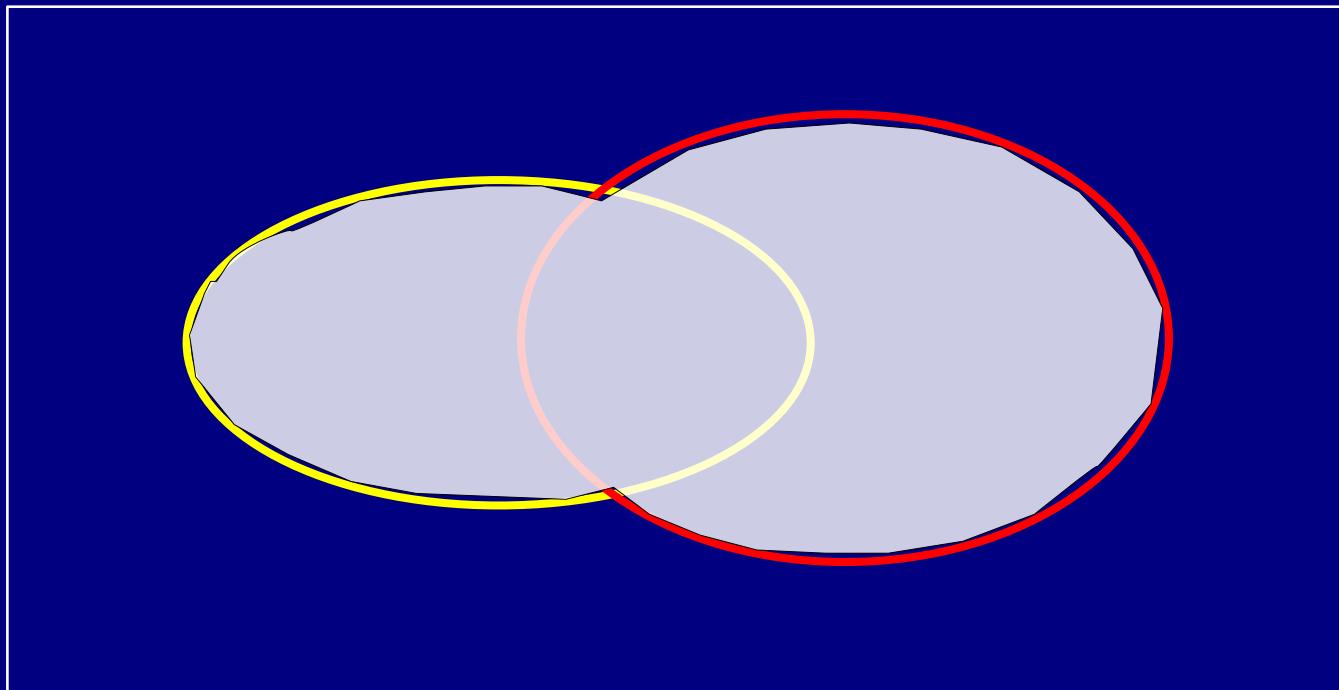
**A**

**$A \cup B$**

**B**

$A \cup B = \text{set of indiv. instances of } A \text{ or of } B$

# OR (Union)

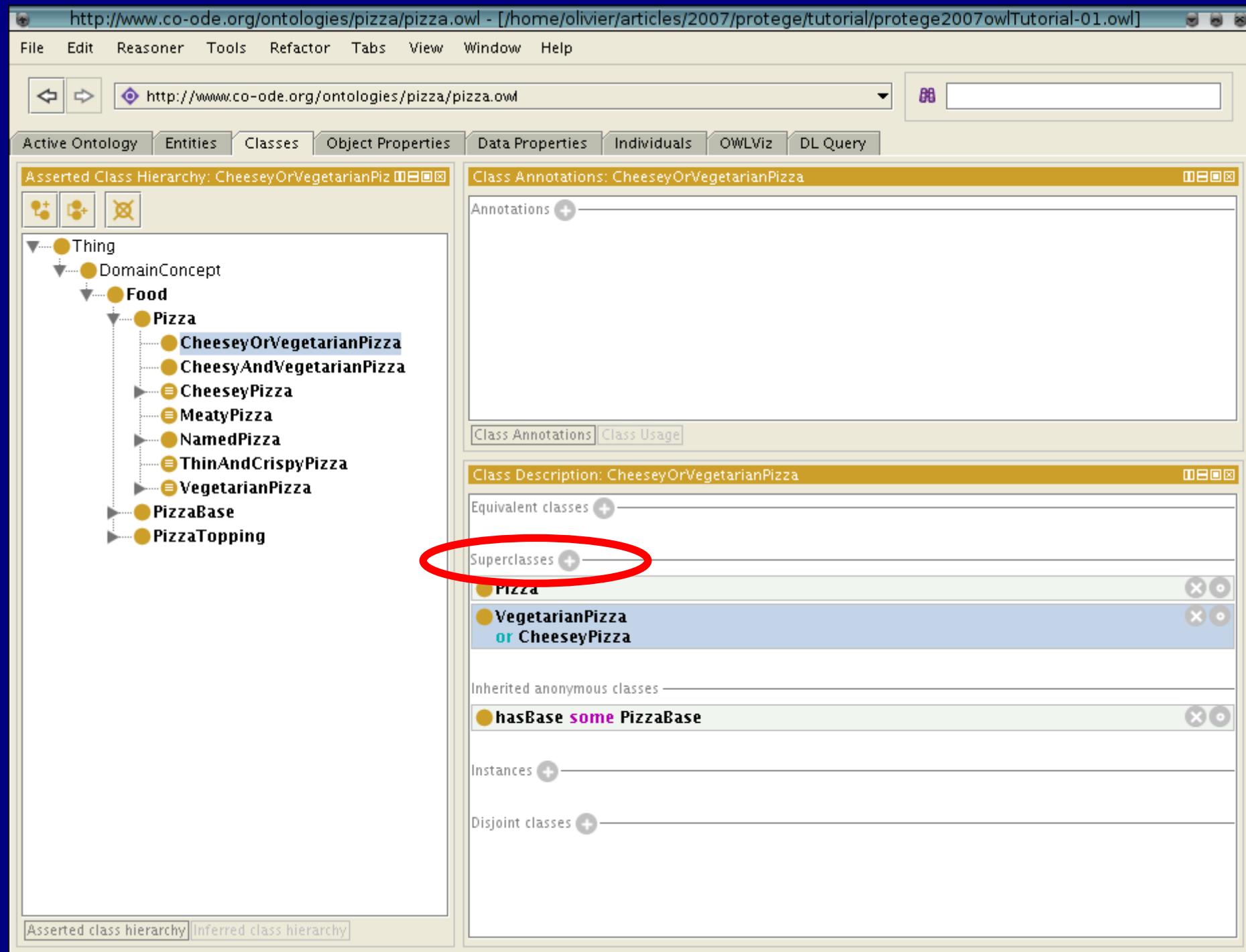


A

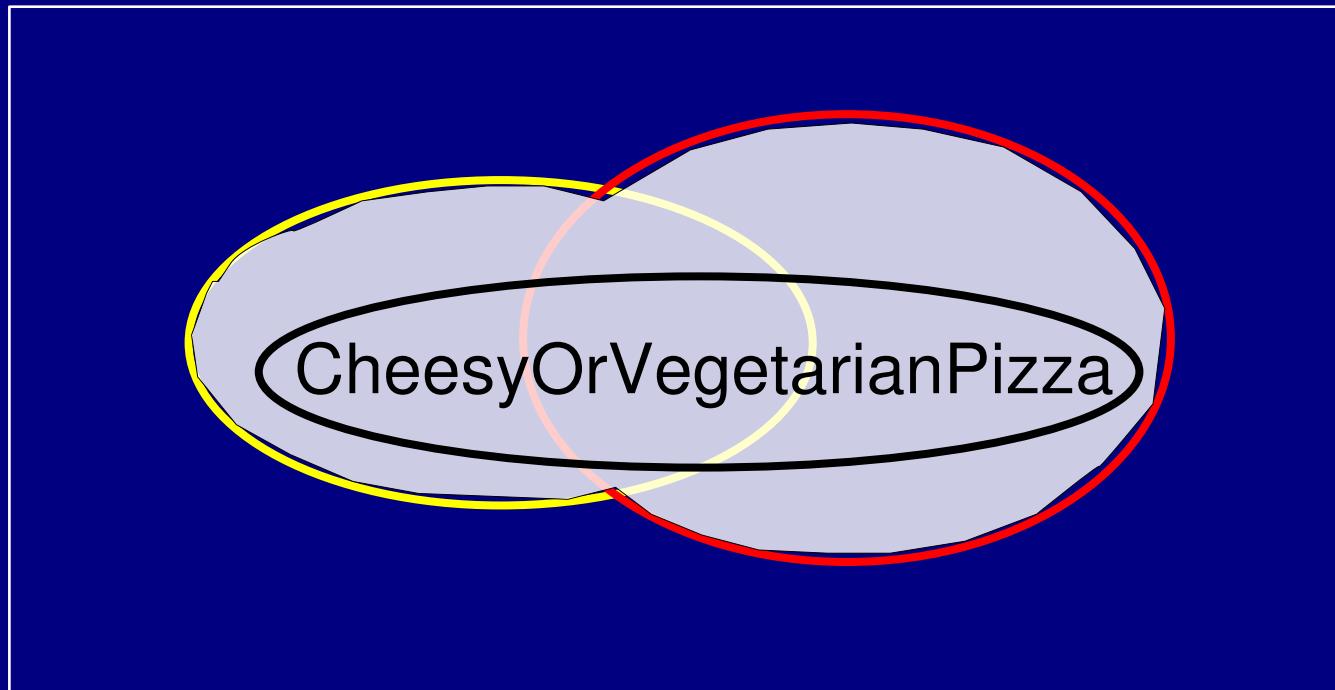
A  $\sqcup$  B

B

Ex: VegetarianPizza  $\sqcup$  CheesyPizza



# OR (Union)



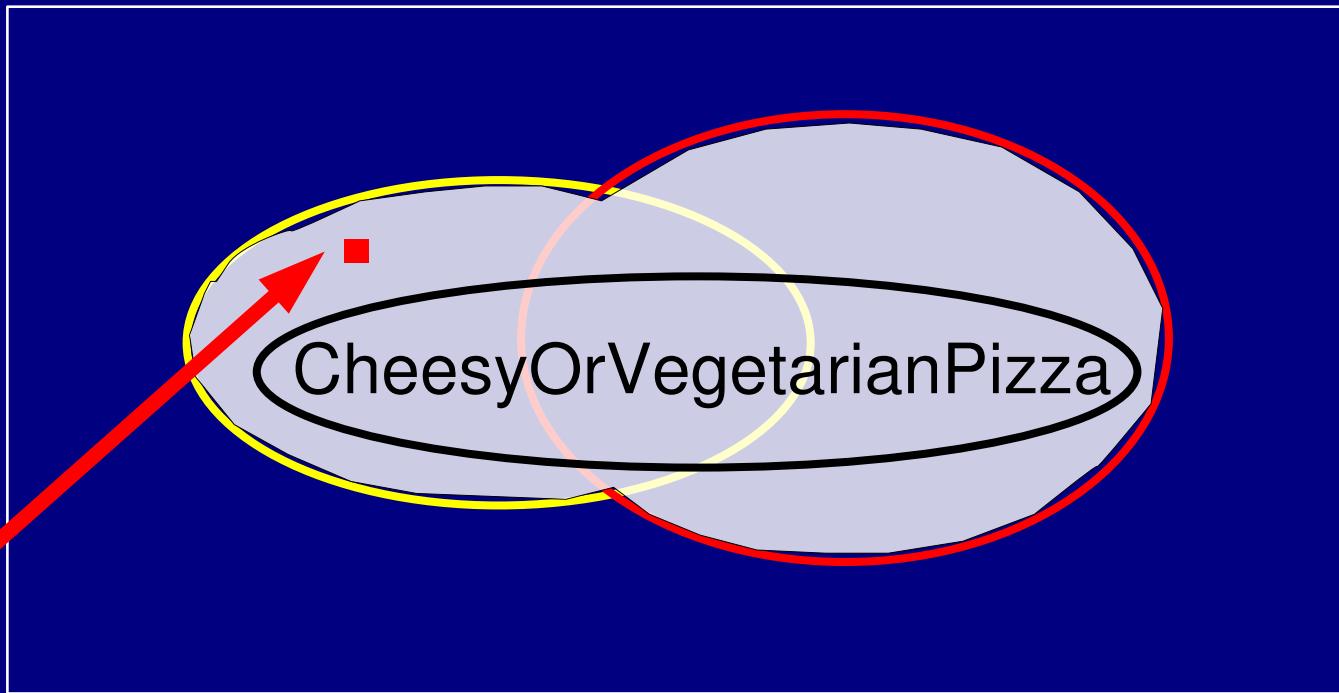
A

A  $\sqcup$  B

B

CheesyOrVegie  $\sqsubset$  VegetarianPizza  $\sqcup$  CheesyPizza

# OR (Union)



- There could be instances of CheesyPizza (red dot) that are not instances of CheesyOrVegetarianPizza...
- ... therefore, CheesyPizza is not subclass of CheesyOrVeggie

# OR (Union)

- Now, use a definition for CheesyOrVegetarianPizza  
(tip: right-click is your friend)

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: CheeseyOrVegetarianPiz

Annotations

Class Annotations: CheeseyOrVegetarianPizza

Class Annotations Class Usage

Class Description: CheeseyOrVegetarianPizza

Equivalent classes

Superclasses

Pizza  
VegetarianPizza  
or CheeseyPizza

Inherited anonymous classes

hasBase some Pizza

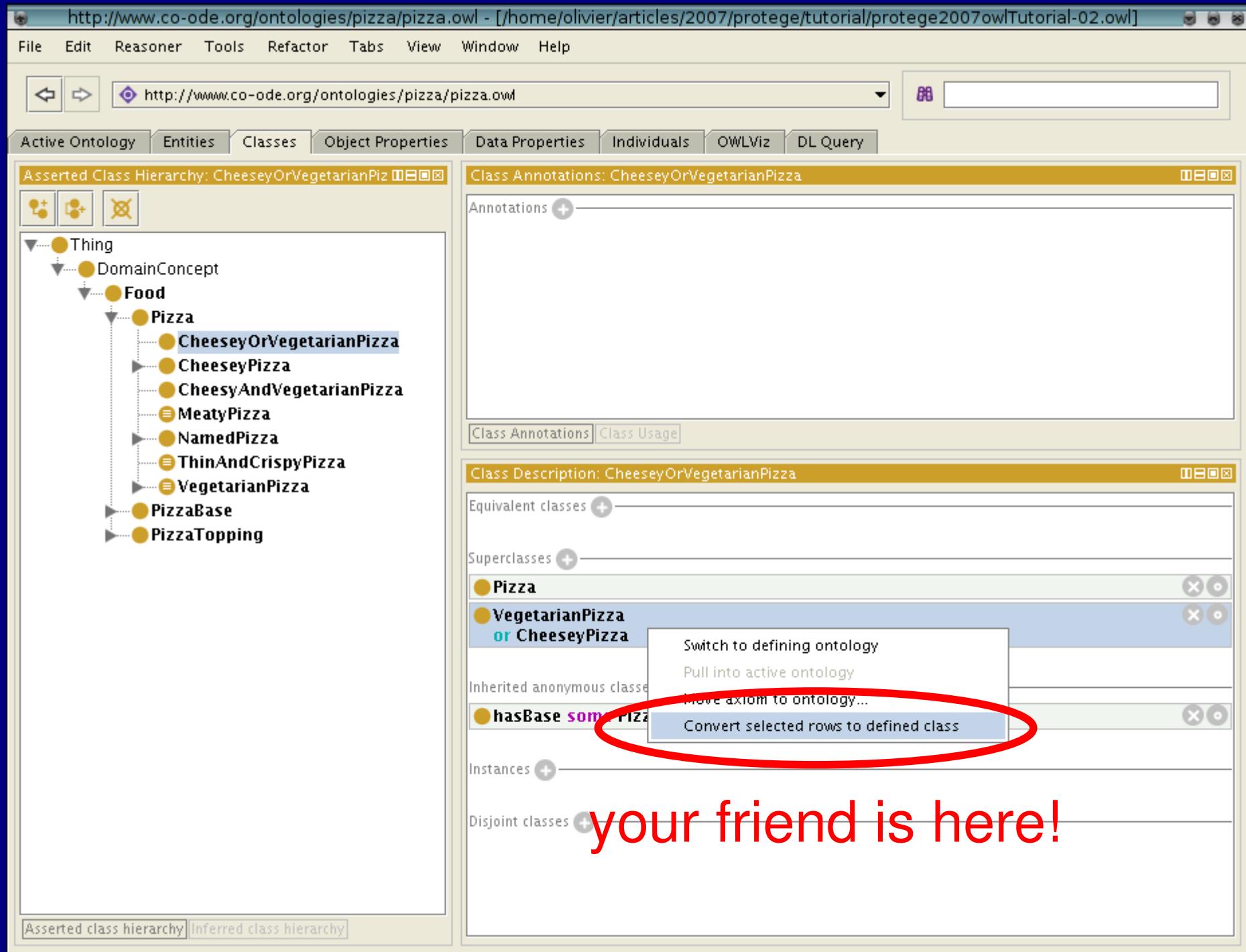
Instances

Disjoint classes

Switch to defining ontology  
Pull into active ontology  
Move axiom to ontology...  
**Convert selected rows to defined class**

your friend is here!

Asserted class hierarchy Inferred class hierarchy



http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Properties

Asserted Class Hierarchy: CheeseyOrVegetarianPiz

Class Annotations

Annotations

Equivalent classes + -

VegetarianPizza  
or CheeseyPizza

Class Annotations Class Usage

Class Description: CheeseyOrVegetarianPizza

Equivalent classes +

VegetarianPizza  
or CheeseyPizza

Superclasses +

Pizza

Inherited anonymous classes -

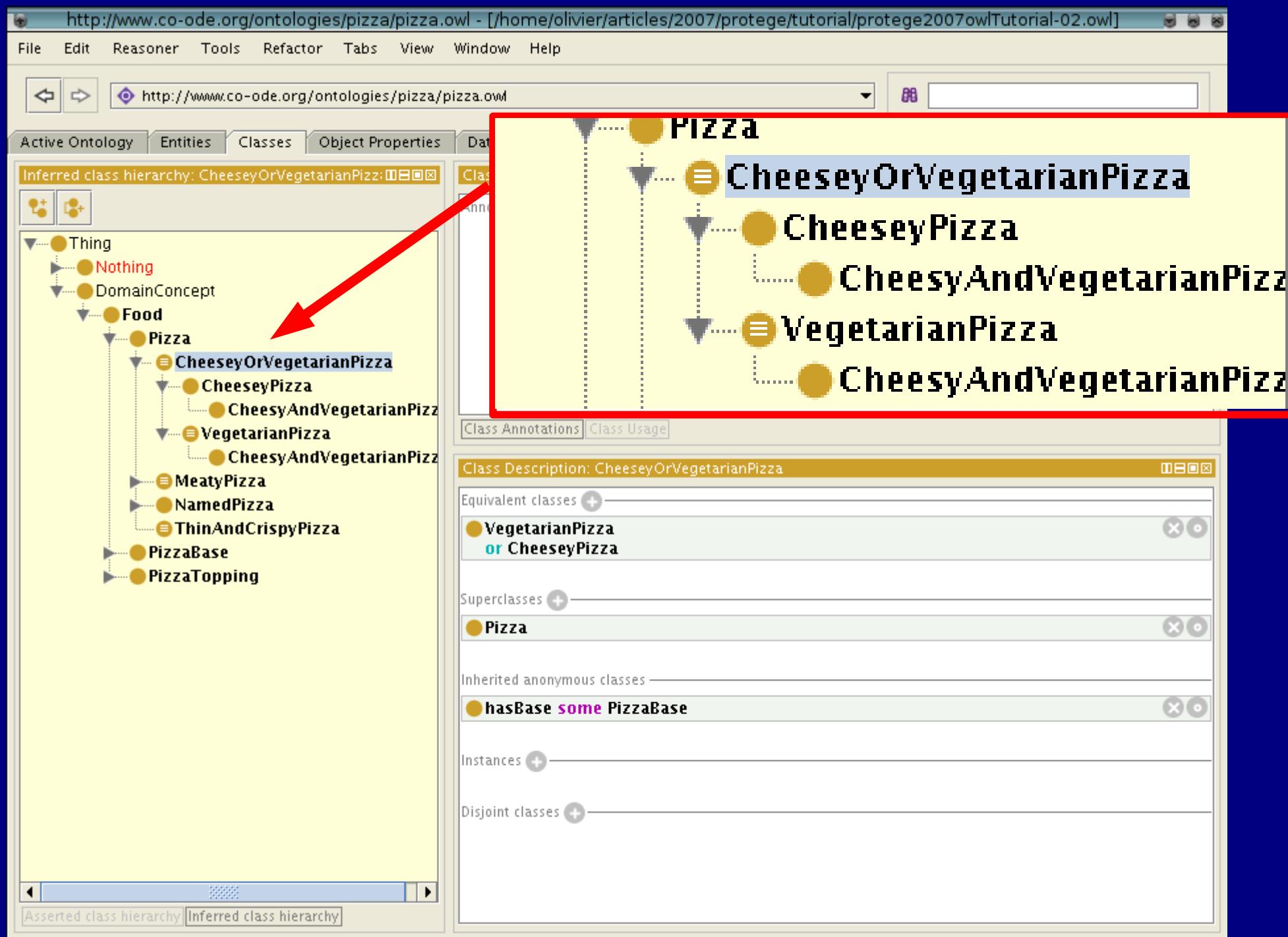
hasBase some PizzaBase

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

The screenshot shows the Protege 2007 interface for editing an ontology. On the left, the 'Asserted Class Hierarchy' tree is displayed, showing the class structure under 'Thing' and 'Food'. The 'CheeseyOrVegetarianPizza' class is highlighted in blue. On the right, the 'Class Description' panel for 'CheeseyOrVegetarianPizza' is open, showing its definition. A red box highlights the 'Equivalent classes' section, which contains the terms 'VegetarianPizza' and 'or CheeseyPizza'. A red arrow points from the 'or' operator in the class description to this section, indicating the connection between the two terms.



# Examples

- Declare MargheritaPizza to be a VegetarianPizza
- Declare AmericanPizza to be a CheesyPizza
- Declare CaprinaPizza to be both CheesyPizza and VegetarianPizza
- Classify
  - :-)
  - why isn't CaprinaPizza classified as expected ?

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: Margherita

- CheeseyPizza
- CheesyAndVegetarianPizza
- MeatyPizza
- NamedPizza
  - American
  - AmericanHot
  - Cajun
  - Capricciosa
  - Caprina
  - Fiorentina
  - FourSeasons
  - FruttiDiMare
  - Giardiniera
  - LaReine
  - Margherita
  - Mushroom
  - Napoletana
  - Parmense
  - PolloAdAstra
  - PrinceCarlo
  - QuattroFormaggi
  - Rosa
  - Siciliana
  - SloppyGiuseppe
  - Soho
  - Veneziana
- ThinAndCrispyPizza
- VegetarianPizza

Class Annotations: Margherita

Annotations +

Superclasses +

- NamedPizza
- VegetarianPizza

Class Description: Margherita

Equivalent classes +

Superclasses +

- NamedPizza
- VegetarianPizza

hasTopping some MozzarellaTopping

hasTopping some TomatoTopping

Inherited anonymous classes –

- hasBase some PizzaBase
- Pizza  
that hasTopping only VegetarianTopping

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

The screenshot illustrates the inheritance of superclasses in an ontology. In the 'Class Annotations' tab, 'Margherita' is shown to inherit from 'NamedPizza' and 'VegetarianPizza'. This inheritance is also reflected in the 'Class Description' tab under the 'Superclasses' section, where 'NamedPizza' and 'VegetarianPizza' are listed as inherited superclasses.

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: American

Annotations

Class Annotations: American

Class Annotations Class Usage

Class Description: American

Equivalent classes

Superclasses

- CheeseyPizza
- NamedPizza
- MeatyPizza

hasTopping some MozzarellaTopping

hasTopping some PeperoniSausageTopping

hasTopping some TomatoTopping

Inherited anonymous classes

- hasBase some PizzaBase
- Pizza  
that hasTopping some MeatTopping

Instances

Asserted class hierarchy Inferred class hierarchy

```
graph TD; American[American] --> CheeseyPizza[CheeseyPizza]; American --> NamedPizza[NamedPizza]; American --> MeatyPizza[MeatyPizza]; American --> AmericanHot[AmericanHot]; American --> Cajun[Cajun]; American --> Capricciosa[Capricciosa]; American --> Caprina[Caprina]; American --> Fiorentina[Fiorentina]; American --> FourSeasons[FourSeasons]; American --> FruttiDiMare[FruttiDiMare]; American --> Giardiniera[Giardiniera]; American --> LaReine[LaReine]; American --> Margherita[Margherita]; American --> Mushroom[Mushroom]; American --> Napoletana[Napoletana]; American --> Parmense[Parmense]; American --> PolloAdAstra[PolloAdAstra]; American --> PrinceCarlo[PrinceCarlo]; American --> QuattroFormaggi[QuattroFormaggi]; American --> Rosa[Rosa]; American --> Siciliana[Siciliana]; American --> SloppyGiuseppe[SloppyGiuseppe]; American --> Soho[Soho]; American --> Veneziana[Veneziana]; American --> ThinAndCrispyPizza[ThinAndCrispyPizza]; American --> VegetarianPizza[VegetarianPizza]
```

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: Caprina

- CheeseyPizza
- CheesyAndVegetarianPizza
- MeatyPizza
- NamedPizza
  - American
  - AmericanHot
  - Cajun
  - Capricciosa
  - Caprina
  - Fiorentina
  - FourSeasons
  - FruttiDiMare
  - Giardiniera
  - LaReine
  - Margherita
  - Mushroom
  - Napoletana
  - Parmense
  - PolloAdAstra
  - PrinceCarlo
  - QuattroFormaggi
  - Rosa
  - Siciliana
  - SloppyGiuseppe
  - Soho
  - Veneziana
- ThinAndCrispyPizza
- VegetarianPizza

Class Annotations: Caprina

Annotations +

Class Annotations Class Usage

Class Description: Caprina

Equivalent classes +

Superclasses +

- NamedPizza
- VegetarianPizza
- and CheeseyPizza

hasTopping some GoatsCheeseTopping

hasTopping some MozzarellaTopping

hasTopping some SundriedTomatoTopping

hasTopping some TomatoTopping

Inherited anonymous classes –

- hasBase some PizzaBase
- Pizza  
that hasTopping only VegetarianTopping

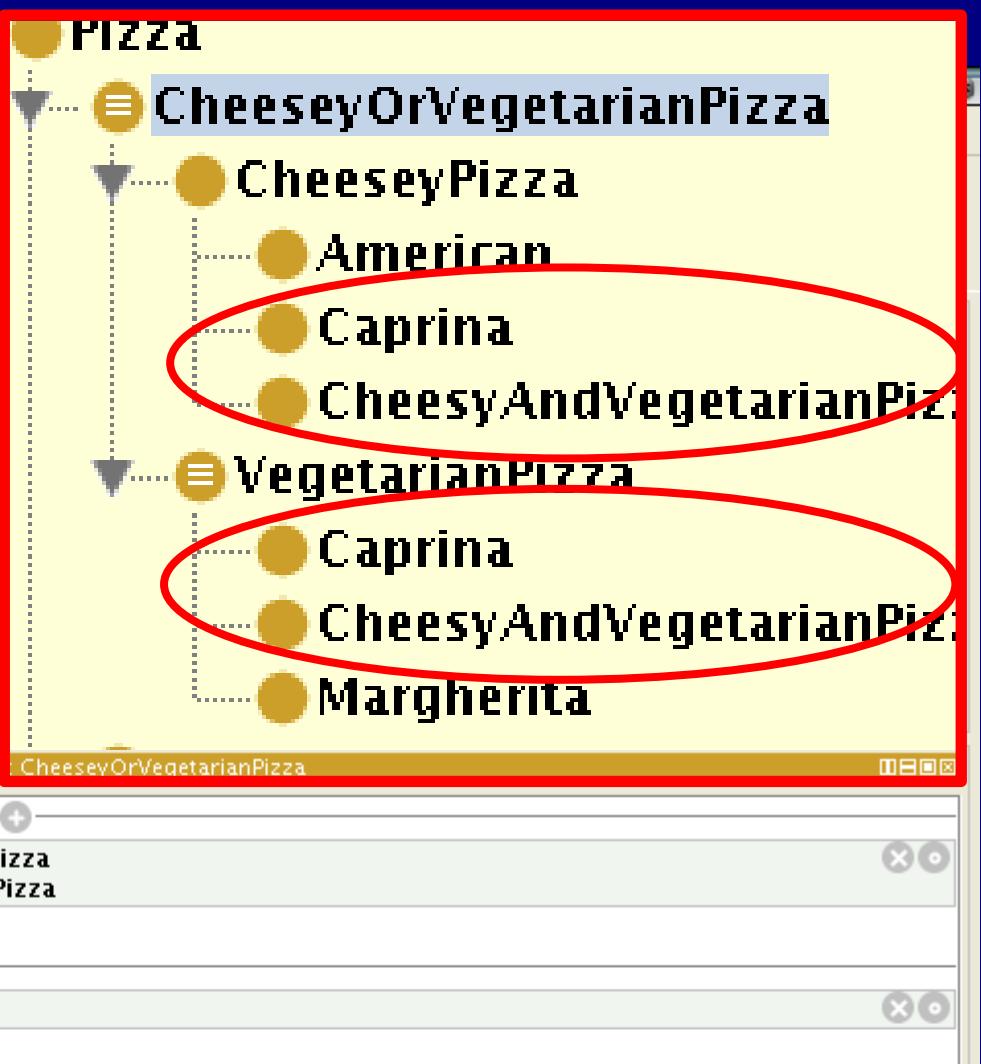
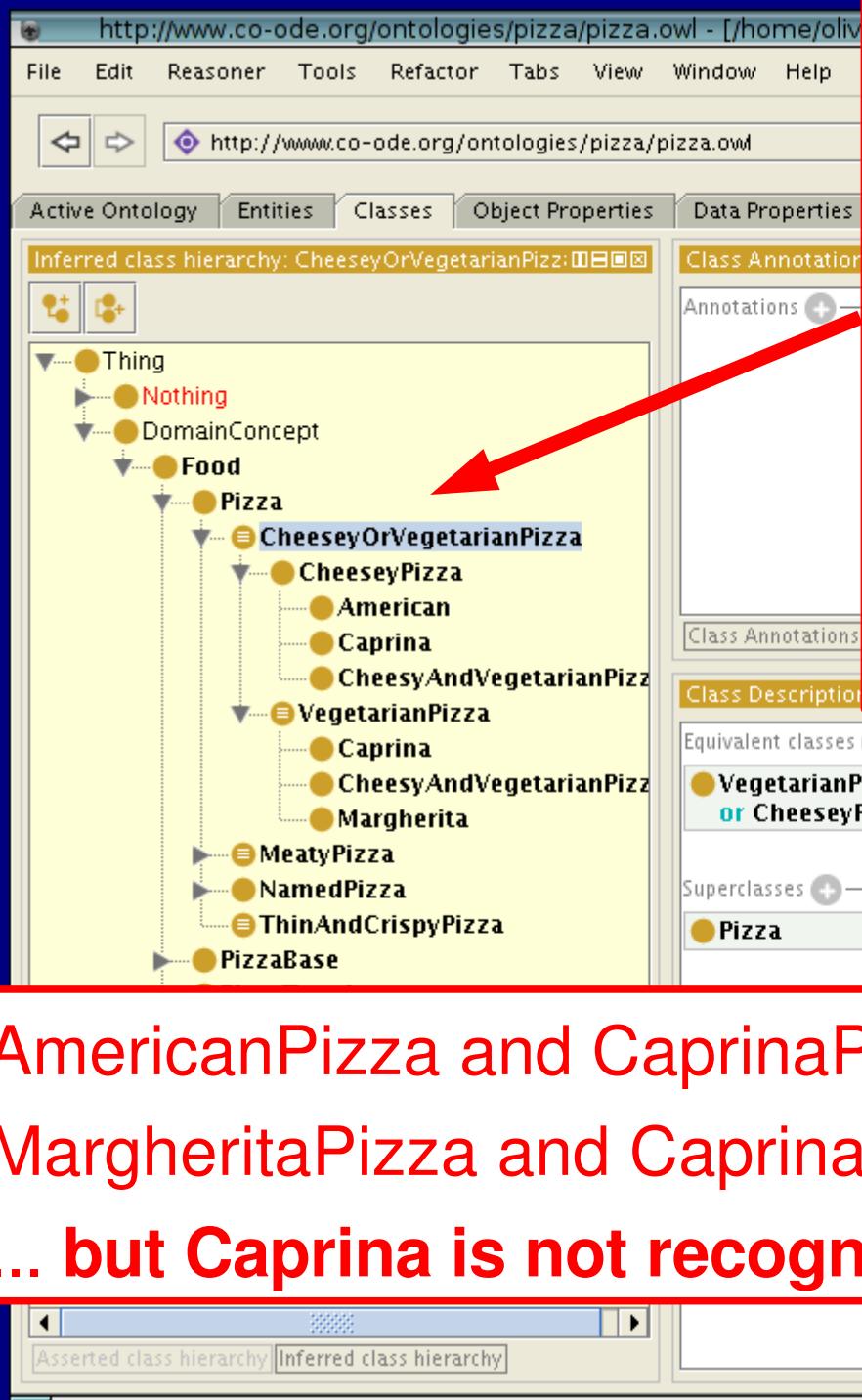
Asserted class hierarchy Inferred class hierarchy

**NamedPizza**

**VegetarianPizza**

**and CheeseyPizza**

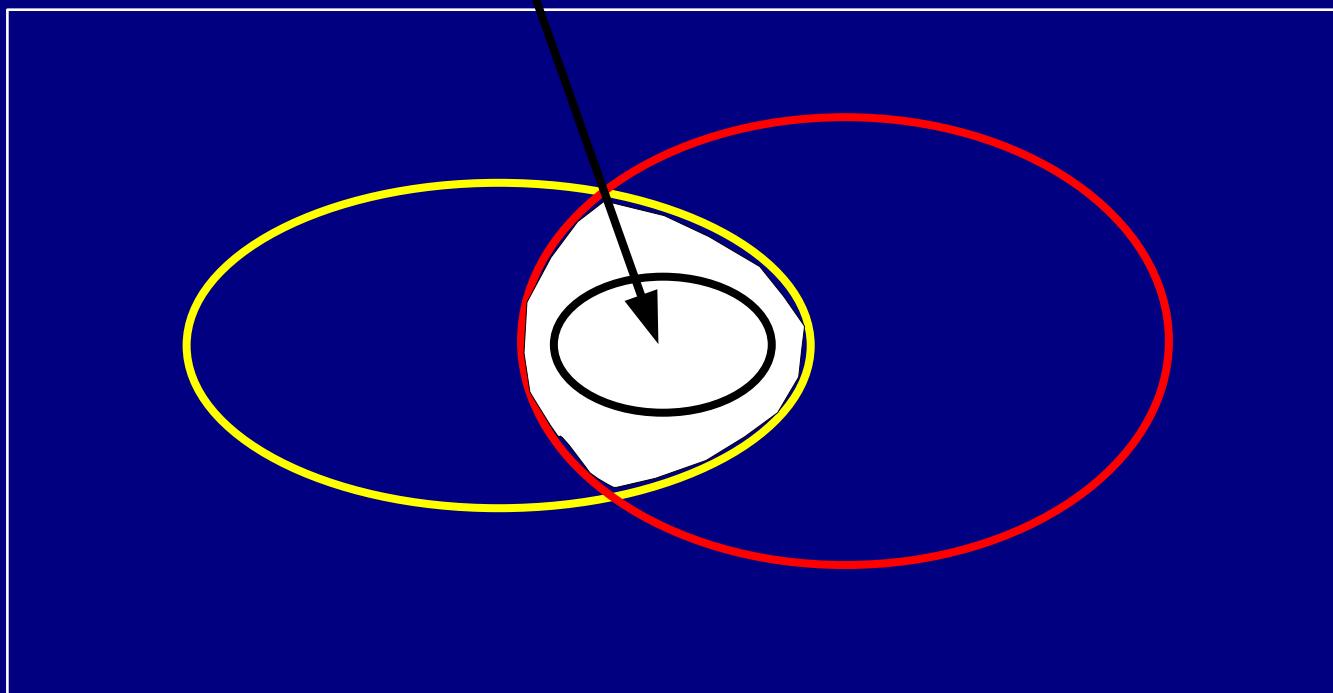
The screenshot shows the Protege ontology editor interface. On the left, the 'Asserted Class Hierarchy' panel displays a tree of pizza classes under 'Caprina'. In the center, the 'Class Annotations' panel shows annotations for 'Caprina'. A red box highlights the 'Annotations' section, which lists 'NamedPizza', 'VegetarianPizza', and 'and CheeseyPizza'. A red arrow points from this box to the 'VegetarianPizza' and 'CheeseyPizza' entries in the class hierarchy tree on the left. The 'Class Description' panel below also lists these three annotations. At the bottom, the 'Inferred class hierarchy' tab is visible.



- AmericanPizza and CaprinaPizza are recognised as cheesey
- MargheritaPizza and CaprinaPizza are recognised as veggie
- ... but Caprina is not recognised as CheesyAndVeggie

# AND (Intersection)

CheesyAndVegetarianPizza



A

A  $\cap$  B

B

Ex: VegetarianPizza  $\cap$  CheesyPizza

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Properties

Asserted Class Hierarchy: CheesyAndVegetarianPizza

Class Annotations

Annotations +

Equivalent classes + -

VegetarianPizza  
and CheeseyPizza

Class Annotations Class Usage

Class Description: CheesyAndVegetarianPizza

Equivalent classes +

VegetarianPizza  
and CheeseyPizza

Superclasses +

Pizza

CheeseyPizza

VegetarianPizza

Inherited anonymous classes –

hasBase some PizzaBase

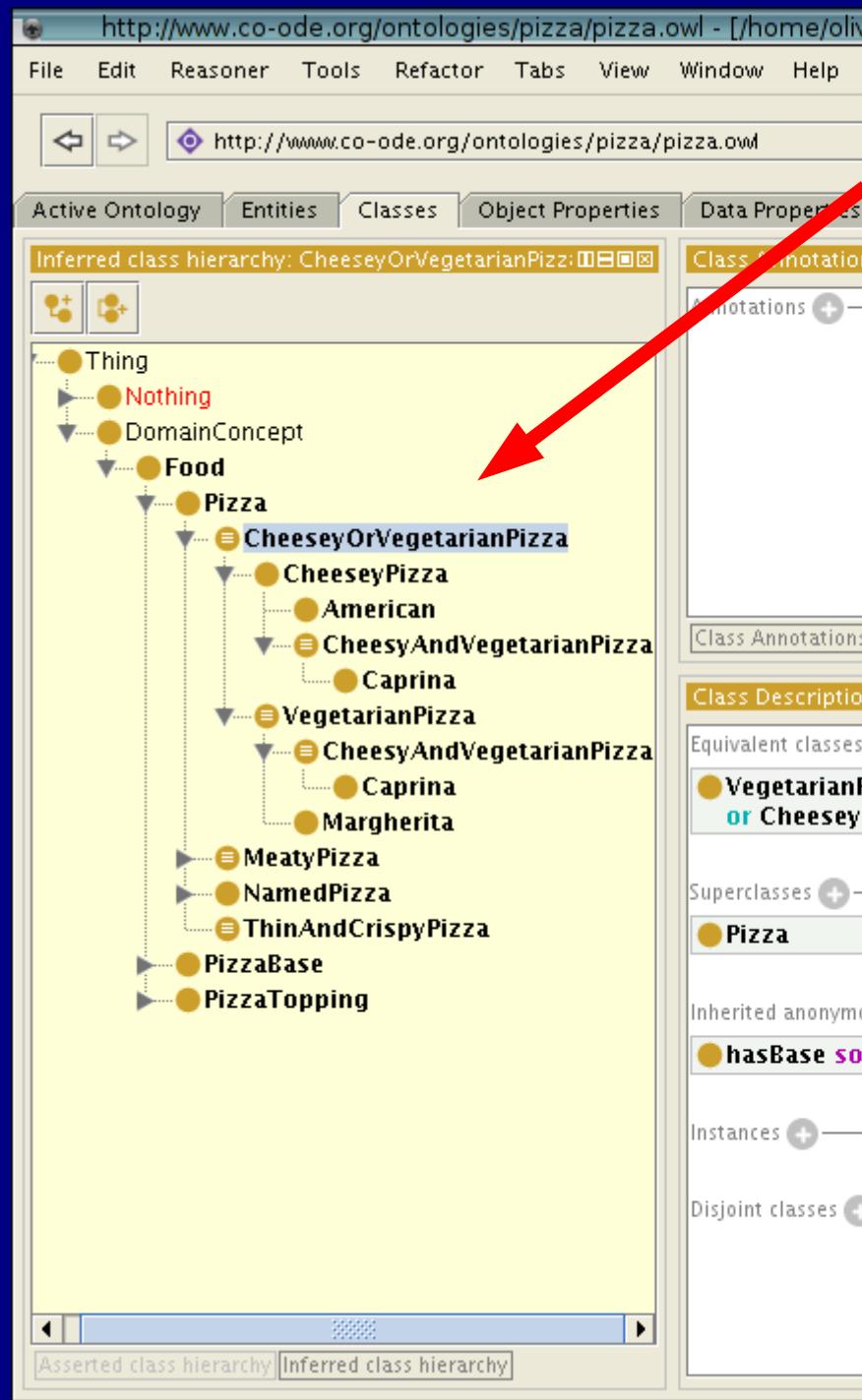
Pizza  
that hasTopping only VegetarianTopping

VegetarianPizza  
or CheeseyPizza

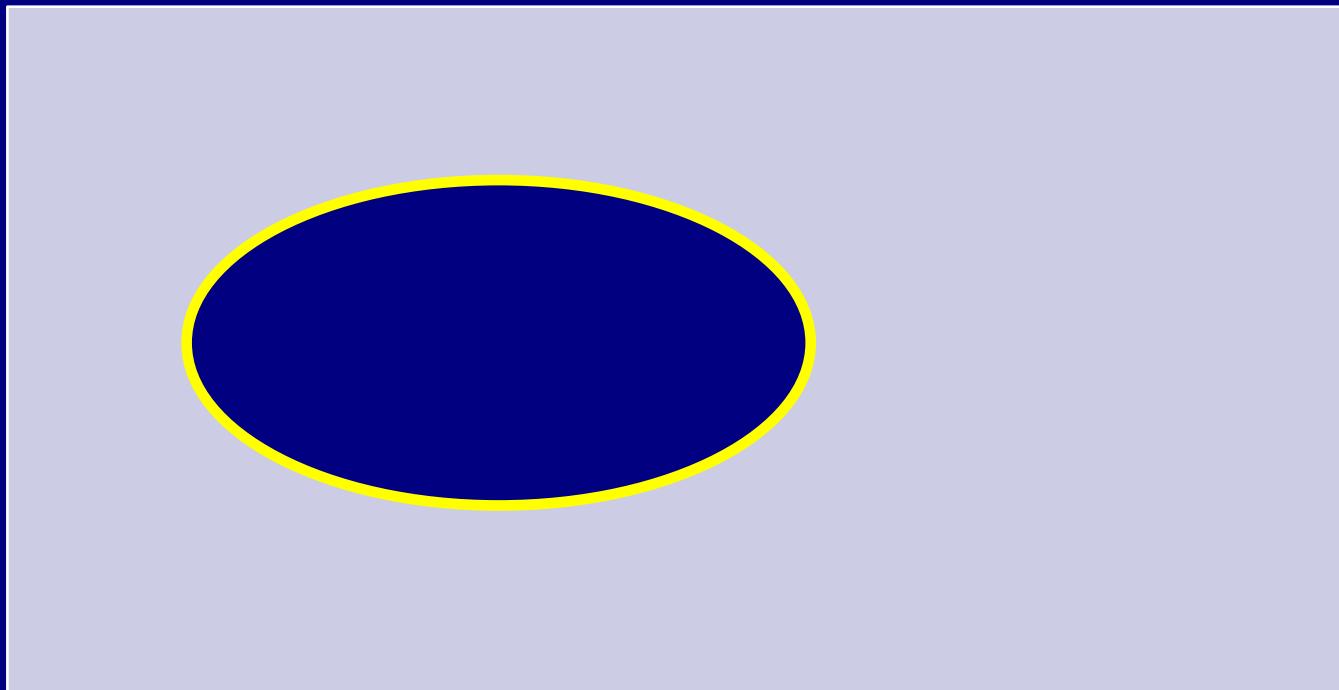
Instances +

Asserted class hierarchy Inferred class hierarchy

The screenshot shows the Protege 2007 interface for editing an ontology. On the left, the 'Asserted Class Hierarchy' tree shows the class structure: Thing > DomainConcept > Food > Pizza > CheeseyOrVegetarianPizza, CheeseyPizza, CheesyAndVegetarianPizza, MeatyPizza, NamedPizza, ThinAndCrispyPizza, VegetarianPizza, PizzaBase, and PizzaTopping. The 'CheesyAndVegetarianPizza' class is selected. On the right, the 'Class Description' panel for 'CheesyAndVegetarianPizza' is open. It contains sections for 'Equivalent classes', 'Superclasses', 'Inherited anonymous classes', and 'Instances'. A red box highlights the 'Equivalent classes' section, and a red arrow points from the 'VegetarianPizza and CheeseyPizza' entry in the 'Class Description' to the same entry in the 'Equivalent classes' section.



# NEGATION (Complement)



A

$\neg A$

Ex:  $\neg$  VegetarianPizza

# NEGATION (Complement)

- Create VegetarianTopping as a subclass of PizzaTopping
- A Vegetarian topping is
  - a topping
  - neither a meat topping, nor a fish topping

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Properties Individual

Asserted Class Hierarchy: VegetarianTopping    Class Annotations: VegetarianTopping

Annotations +

Class Annotations Class Usage

Superclasses +

PizzaTopping

not FishTopping

not MeatTopping

Class Description: VegetarianTopping

Equivalent classes +

Superclasses +

PizzaTopping

not FishTopping

not MeatTopping

Inherited anonymous classes

Instances +

Disjoint classes +

Thing

DomainConcept

Food

Pizza

- CheeseyOrVegetarianPizza
- CheeseyPizza
- CheesyAndVegetarianPizza
- MeatyPizza
- NamedPizza
- ThinAndCrispyPizza
- VegetarianPizza

PizzaBase

PizzaTopping

- VegetarianTopping
- VegetarianToppingNick
- CheeseTopping
- FishTopping
- FruitTopping
- HerbSpiceTopping
- MeatTopping
- NutTopping
- SauceTopping
- VegetableTopping

Asserted class hierarchy Inferred class hierarchy

```
graph TD; VT[VegetarianTopping] --> PT[PizzaTopping]; VT --> NFT[not FishTopping]; VT --> NMT[not MeatTopping];
```

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Inferred class hierarchy: VegetarianTopping

Annotations

Class Annotations Class Usage

Class Description: VegetarianTopping

Equivalent classes

Superclasses

PizzaTopping  
not FishTopping  
not MeatTopping

Disjoint classes

Asserted class hierarchy Inferred class hierarchy

VegetableTopping, FruitTopping, ... are not recognised as VegetarianToppings

# NEGATION (Complement)

- Create VegetarianTopping as a subclass of PizzaTopping
- A Vegetarian topping is neither a meat topping, nor a fish topping
- Classify
- Why do we have to provide a Necessary and Sufficient definition ?

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies/pizza/pizza...

Active Ontology Entities Classes Object Properties Data

Asserted Class Hierarchy: VegetarianTopping

Class Annotations Class Usage

Class Description: VegetarianTopping

Equivalent classes +

PizzaTopping  
and not MeatTopping  
and not FishTopping

Class Annotations Class Usage

Class Description: VegetarianTopping

Equivalent classes +

PizzaTopping  
and not MeatTopping  
and not FishTopping

Superclasses +

PizzaTopping

Inherited anonymous classes —

Instances +

Disjoint classes +

Thing

DomainConcept

Food

Pizza

- CheeseyOrVegetarianPizza
- CheeseyPizza
- CheesyAndVegetarianPizza
- MeatyPizza
- NamedPizza
- ThinAndCrispyPizza
- VegetarianPizza

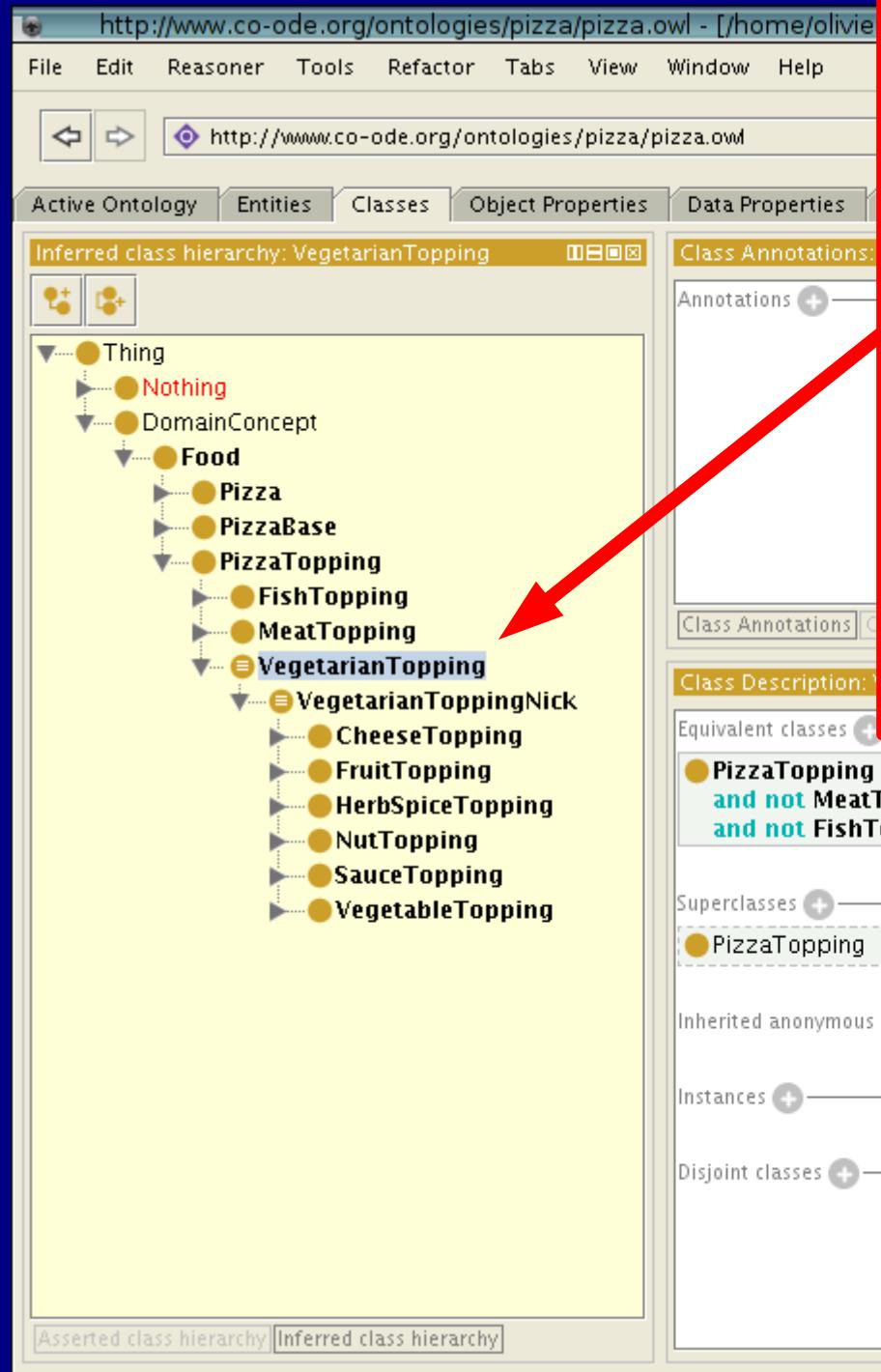
PizzaBase

PizzaTopping

- VegetarianTopping
- VegetarianToppingNick
- CheeseTopping
- FishTopping
- FruitTopping
- HerbSpiceTopping
- MeatTopping
- NutTopping
- SauceTopping
- VegetableTopping

Asserted class hierarchy Inferred class hierarchy

A red circle highlights the "Equivalent classes" button above the main class description area. A red arrow points from the "Equivalent classes" section in the main class description area down to the "Equivalent classes" section in the detailed view.



# NEGATION (Complement)

- A Vegetarian topping is neither a meat topping, nor a fish topping
- Why do we have to provide a Necessary and Sufficient definition ?
  - it ensures that all the instances of PizzaTopping that are neither instances of MeatTopping nor of FishTopping are inferred to be instances of VegetarianTopping

# NEGATION (Complement)

- Create VegetarianTopping as a subclass of PizzaTopping
- A Vegetarian topping is neither a meat topping, nor a fish topping
- Why do we have to provide a N&S definition ?
- Create NonVegetarianTopping
- Classify

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data

Asserted Class Hierarchy: NonVegetarianTopping

Class Annotations Class Usage

Class Description: NonVegetarianTopping

Equivalent classes +

PizzaTopping  
and not VegetarianTopping

Superclasses +

Inherited anonymous classes -

Instances +

Disjoint classes +

Thing

DomainConcept

Food

Pizza

CheeseyOrVegetarianPizza

CheeseyPizza

CheesyAndVegetarianPizza

MeatyPizza

NamedPizza

ThinAndCrispyPizza

VegetarianPizza

PizzaBase

PizzaTopping

NonVegetarianTopping

VegetarianTopping

VegetarianToppingNick

CheeseTopping

FishTopping

FruitTopping

HerbSpiceTopping

MeatTopping

NutTopping

SauceTopping

VegetableTopping

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data P...  
Inferred class hierarchy: NonVegetarianTopping

Class A  
Annotations  
NonVegetarianTopping  
FishTopping  
MeatTopping

Class Annotations Class Usage

Class Description: NonVegetarianTopping

Equivalent classes +  
**PizzaTopping  
and not VegetarianTopping**

Superclasses +  
**PizzaTopping**

Inherited anonymous classes -

Instances +

Disjoint classes +

PIZZATopping

NonVegetarianTopping

FishTopping

MeatTopping

Nothing

DomainConcept

Food

Pizza

PizzaBase

PizzaTopping

NonVegetarianTopping

FishTopping

MeatTopping

VegetarianTopping

VegetarianToppingNick

CheeseTopping

FruitTopping

HerbSpiceTopping

NutTopping

SauceTopping

VegetableTopping

Asserted class hierarchy Inferred class hierarchy

```
graph TD; NonVegetarianTopping[NonVegetarianTopping] --> FishTopping[FishTopping]; NonVegetarianTopping --> MeatTopping[MeatTopping]; NonVegetarianTopping --> PizzaTopping[PizzaTopping]
```

# NEGATION (Complement)

- Note that the reasoner found out that CheeseTopping and VegetableTopping are subclasses of VegetarianTopping whereas the definition of VegetarianTopping does not mention CheeseTopping nor VegetableTopping (intentionality)

# Expressing constraints

# Objective

Application of the intensional approach: leverage the expressivity of the OWL-DL language for a precise representation of the classes' features

- We will describe the pizza ingredients and use the reasoner to find out which one are cheesy and/or vegetarian

# Getting in sync!

If you need to catch-up,  
the ontology at this point is  
protege2007owlTutorial-02.owl

from:

<http://www.ea3888.univ-rennes1.fr/dameron/protege2007/>

# Constraints

## 1. Quantifier restriction (at least one, all of)

- How to represent the fact that every pizza must have *at least* a topping ?
- How to represent the fact that *all* the ingredients of a vegetarian pizza must be vegetarians ?

## 2. Cardinality restrictions

- How to represent that a Hand must have *5* fingers as parts ?

## 3. hasValue restrictions

- How to define the value of a relation for a class ?

# Principles

- A restriction describes an anonymous class composed of all the individuals that satisfy the restriction
  - e.g. all the individuals that have (amongst other things) mozzarella as topping
- This anonymous class is used as a superclass of the (named) class we want to express a constraint on
  - e.g. MargheritaPizza

# Existential restriction

- $(\exists \text{ } hasTopping \text{ Mozzarella})$  : set of the individuals being linked to at least one instance of Mozzarella through the *hasTopping* property
  - They can be linked to multiple instances of Mozzarella
  - They can also be linked to instances of other classes (provided domain and range integrity)
  - Margherita  $\sqsubset (\exists \text{ } hasTopping \text{ Mozzarella})$

# Existential restriction

- $(\exists \text{ } hasTopping \text{ Mozzarella})$  : set of the individuals being linked to at least one instance of Mozzarella through the *hasTopping* property
  - They can be linked to multiple instances of Mozzarella
  - They can also be linked to instances of other classes (provided domain and range integrity)
- Margherita  $\sqsubset (\exists \text{ } hasTopping \text{ Mozzarella})$ 
  - Other pizza can also have Mozzarella!

# Complete the ontology

- Define CheesyPizza as a pizza having at least one cheese topping
- Remove the fact that AmericanPizza and CaprinaPizza are subclasses of CheesyPizza !

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies

Active Ontology Entities Classes Object Prop

Asserted Class Hierarchy: CheeseyPizza

Thing  
DomainConcept  
Food  
Pizza  
CheeseyOrVegetarianPizza  
CheeseyPizza  
CheesyAndVegetarianPizza  
MeatyPizza  
NamedPizza  
ThinAndCrispyPizza  
VegetarianPizza  
PizzaBase  
PizzaTopping  
NonVegetarianTopping  
VegetarianTopping  
VegetarianToppingNick  
CheeseTopping  
FishTopping  
FruitTopping  
HerbSpiceTopping  
MeatTopping  
NutTopping  
SauceTopping  
VegetableTopping

Equivalent classes +

Pizza  
that hasTopping some CheeseTopping

Class Annotations Class Usage

Class Description: CheeseyPizza

Equivalent classes +

Pizza  
that hasTopping some CheeseTopping

Superclasses +

Pizza  
CheeseyOrVegetarianPizza

Inherited anonymous classes –

hasBase some PizzaBase

VegetarianPizza  
or CheeseyPizza

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

The image shows a screenshot of the Protege 2007 OWL tutorial interface. On the left, there's a tree view of the asserted class hierarchy under the 'CheeseyPizza' class. The right side features a detailed view of the 'CheeseyPizza' class itself. A red box highlights the 'Equivalent classes' tab, which contains the description 'Pizza that hasTopping some CheeseTopping'. A red arrow points from this highlighted area down to the 'Equivalent classes' section of the main class description panel below. The main class description panel also lists 'Superclasses' (Pizza, CheeseyOrVegetarianPizza), 'Inherited anonymous classes' (hasBase some PizzaBase, VegetarianPizza or CheeseyPizza), and sections for 'Instances' and 'Disjoint classes'.

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: American

Pizza

- CheeseyOrVegetarianPizza
- CheeseyPizza
- CheesyAndVegetarianPizza
- MeatyPizza
- NamedPizza
- American
- AmericanHot
- Cajun
- Capricciosa
- Caprina
- Fiorentina
- FourSeasons
- FruttiDiMare
- Giardiniera
- LaReine
- Margherita
- Mushroom
- Napoletana
- Parmense
- PolloAdAstra
- PrinceCarlo
- QuattroFormaggi
- Rosa
- Siciliana
- SloppyGiuseppe
- Soho
- Veneziana

Annotations

Class Annotations Class Usage

Class Description: American

Equivalent classes

Superclasses

- NamedPizza
- hasTopping some MozzarellaTopping
- hasTopping some PeperoniSausageTopping
- hasTopping some TomatoTopping
- MeatyPizza

Inherited anonymous classes

- hasBase some PizzaBase
- Pizza  
that hasTopping some MeatTopping

Instances

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: Caprina

Pizza

- CheeseyOrVegetarianPizza
- CheeseyPizza
- CheesyAndVegetarianPizza
- MeatyPizza
- NamedPizza
- American
- AmericanHot
- Cajun
- Capricciosa
- Caprina
- Fiorentina
- FourSeasons
- FruttiDiMare
- Giardiniera
- LaReine
- Margherita
- Mushroom
- Napoletana
- Parmense
- PolloAdAstra
- PrinceCarlo
- QuattroFormaggi
- Rosa
- Siciliana
- SloppyGiuseppe
- Soho
- Veneziana

Annotations

Class Annotations Class Usage

Class Description: Caprina

Equivalent classes

Superclasses

- NamedPizza
- hasTopping some GoatsCheeseTopping
- hasTopping some MozzarellaTopping
- hasTopping some SundriedTomatoTopping
- hasTopping some TomatoTopping

Inherited anonymous classes

- hasBase some PizzaBase

Instances

Disjoint classes

- American

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Inferred class hierarchy: CheeseyPizza

Annotations

comment

Any pizza that has at least 1 cheese topping.@en

Class Annotations Class Usage

Class Description: CheeseyPizza

Equivalent classes

Pizza  
that hasTopping some CheeseTopping

Superclasses

Pizza  
CheeseyOrVegetarianPizza

Inherited anonymous classes

hasBase some PizzaBase

VegetarianPizza  
or CheeseyPizza

Instances

Disjoint classes

**CheeseyOrVegetarianPizza**

**CheeseyPizza**

American

AmericanHot

Cajun

Capricciosa

Caprina

**CheesyAndVegetarianP**

Margherita

Fiorentina

FourSeasons

Giardiniera

LaReine

Mushroom

Napoletana

Parmense

PolloAdAstra

PrinceCarlo

QuattroFormaggi

Rosa

Siciliana

SloppyGiuseppe

Soho

Veneziana

VegetarianPizza

MeatyPizza

Asserted class hierarchy Inferred class hierarchy

```
graph TD; Pizza --> CheeseyOrVegetarianPizza; CheeseyOrVegetarianPizza --> CheeseyPizza; CheeseyPizza --> American; CheeseyPizza --> AmericanHot; CheeseyPizza --> Cajun; CheeseyPizza --> Capricciosa; CheeseyPizza --> Caprina; CheeseyPizza --> CheesyAndVegetarianP; CheesyAndVegetarianP --> Margherita; CheesyAndVegetarianP --> Fiorentina; CheesyAndVegetarianP --> FourSeasons; CheesyAndVegetarianP --> Giardiniera; CheesyAndVegetarianP --> LaReine; CheesyAndVegetarianP --> Mushroom; CheesyAndVegetarianP --> Napoletana; CheesyAndVegetarianP --> Parmense; CheesyAndVegetarianP --> PolloAdAstra; CheesyAndVegetarianP --> PrinceCarlo; CheesyAndVegetarianP --> QuattroFormaggi; CheesyAndVegetarianP --> Rosa; CheesyAndVegetarianP --> Siciliana; CheesyAndVegetarianP --> SloppyGiuseppe; CheesyAndVegetarianP --> Soho; CheesyAndVegetarianP --> Veneziana; CheesyAndVegetarianP --> VegetarianPizza; VegetarianPizza --> MeatyPizza;
```

# Universal restriction

- $(\forall \text{ } \textit{hasTopping} \text{ VegetarianTopping})$  : set of all the individuals only linked to instances of VegetarianTopping through the *hasTopping* property
- Warning: also includes all the individuals linked to nothing through the *hasTopping* property

# Universal restriction

- $(\forall \text{ hasTopping} \text{ VegetarianTopping})$
- Remove the fact that MargheritaPizza and CaprinaPizza are subclasses of VegetarianPizza
- Define VegetarianPizza as any pizza for which all the toppings are vegetarian toppings
- Classify :- (

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties

Asserted Class Hierarchy: VegetarianPizza

Thing  
DomainConcept  
Food  
Pizza  
CheeseyOrVegetarianPizza  
CheeseyPizza  
CheesyAndVegetarianPizza  
CheesyAndVegetarianPizza  
MeatyPizza  
NamedPizza  
ThinAndCrispyPizza  
VegetarianPizza  
CheesyAndVegetarianPizza  
PizzaBase  
PizzaTopping  
NonVegetarianTopping  
VegetarianTopping  
VegetarianToppingNick  
CheeseTopping  
FishTopping  
FruitTopping  
HerbSpiceTopping  
MeatTopping  
NutTopping  
SauceTopping  
VegetableTopping

Equivalent classes +

Pizza  
that hasTopping only VegetarianTopping

VegetarianPizza because PizzaTopping is not covering@en

Class Annotations Class Usage

Class Description: VegetarianPizza

Equivalent classes +

Pizza  
that hasTopping only VegetarianTopping

Superclasses +

CheeseyOrVegetarianPizza

Inherited anonymous classes –

hasBase some PizzaBase  
VegetarianPizza  
or CheeseyPizza

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

A red box highlights the 'Equivalent classes' section for the 'VegetarianPizza' class, which contains the description 'Pizza that hasTopping only VegetarianTopping'. A red arrow points from this section down to the 'Class Description' section for 'VegetarianPizza', which also displays the same description.

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Inferred class hierarchy: VegetarianPizza

Annotations

comment

Any pizza that only has vegetarian toppings or no toppings is a VegetarianPizzaEquiv1. Should be inferred to be equivalent to VegetarianPizzaEquiv2. Not equivalent to VegetarianPizza because PizzaTopping is not covering@en

Class Annotations: VegetarianPizza

Class Description: VegetarianPizza

Equivalent classes

Pizza  
that hasTopping only VegetarianTopping

Superclasses

CheeseyOrVegetarianPizza

Inherited anonymous classes

hasBase some PizzaBase  
VegetarianPizza  
or CheeseyPizza

Instances

Disjoint classes

:-)

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Inferred class hierarchy: VegetarianPizza

Annotations

comment

Any pizza that only has vegetarian toppings or no toppings is a VegetarianPizzaEquiv1. Should be inferred to be equivalent to VegetarianPizzaEquiv2. Not equivalent to VegetarianPizza because PizzaTopping is not covering@en

Class Annotations: VegetarianPizza

Class Description: VegetarianPizza

Equivalent classes

Pizza  
that hasTopping only VegetarianTopping

Superclasses

CheeseyOrVegetarianPizza

Inherited anonymous classes

hasBase some PizzaBase

VegetarianPizza  
or CheeseyPizza

Instances

Disjoint classes

Thing

Nothing

DomainConcept

Food

Pizza

CheeseyOrVegetarianPizza

Cheeseypizza

VegetarianPizza

CheesyAndVegetarianPizz

MartyPizza

NamedPizza

ThinAndCrispyPizza

PizzaBase

PizzaTopping

NewvegetarianTopping

VegetarianTopping

VegetarianToppingNick

CheeseTopping

FruitTopping

HerbSpiceTopping

NutTopping

SauceTopping

VegetableTopping

Asserted class hierarchy Inferred class hierarchy

# Universal restriction

- Why Margherita and Caprina pizza were not recognised as vegetarian pizza?  
(even though the vegetarian toppings were correctly recognised)
- ... find out in a few slides

# Cardinality restriction

- PizzaWithTwoToppings
  - $\text{Pizza} \sqcap (\text{hasTopping} = 2)$
- PizzaWithFiveOrMoreToppings
  - $\text{Pizza} \sqcap (\text{hasTopping} \geq 5)$
- PizzaWithThreeOrLessToppings
  - $\text{Pizza} \sqcap (\text{hasTopping} \leq 3)$
- Warning: This is NOT qualified cardinality restr.

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties

Asserted Class Hierarchy: PizzaWithTwoToppings

Equivalent classes +

Pizza  
that hasTopping exactly 2 Thing

Class Annotations Class Usage

Class Description: PizzaWithTwoToppings

Equivalent classes +

Pizza  
that hasTopping exactly 2 Thing

Superclasses +

Pizza

Inherited anonymous classes –

hasBase some PizzaBase

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

The screenshot shows the Protege 2007 interface with the ontology file 'pizza.owl' loaded. The left pane displays the 'Asserted Class Hierarchy' for the class 'PizzaWithTwoToppings', which is a subclass of 'Food'. The hierarchy includes various pizza types like 'CheeseyOrVegetarianPizza', 'MeatyPizza', and 'NamedPizza'. Below this is the 'PizzaTopping' class with subtypes such as 'NonVegetarianTopping', 'VegetarianTopping', and 'SauceTopping'. The right pane is focused on the 'Equivalent classes' tab for the 'PizzaWithTwoToppings' class. It contains the class definition: 'Pizza that hasTopping exactly 2 Thing'. Below this, under 'Superclasses', is the general class 'Pizza'. Other tabs visible include 'Class Annotations' and 'Class Usage'. The bottom navigation bar offers options for 'Asserted class hierarchy' and 'Inferred class hierarchy'.

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties

Asserted Class Hierarchy: PizzaWithFiveOrMoreToppings

Thing  
DomainConcept  
Food  
Pizza  
  PizzaWithThreeOrLessToppings  
  PizzaWithFiveOrMoreToppings  
  PizzaWithTwoToppings  
  CheeseyOrVegetarianPizza  
  CheeseyPizza  
  CheesyAndVegetarianPizza  
  MeatyPizza  
  NamedPizza  
  ThinAndCrispyPizza  
  VegetarianPizza  
  PizzaBase  
  PizzaTopping  
    NonVegetarianTopping  
    VegetarianTopping  
    VegetarianToppingNick  
    CheeseTopping  
    FishTopping  
    FruitTopping  
    HerbSpiceTopping  
    MeatTopping  
    NutTopping  
    SauceTopping  
    VegetableTopping

Equivalent classes +

Pizza  
that hasTopping min 5 Thing

Class Annotations Class Usage

Class Description: PizzaWithFiveOrMoreToppings

Equivalent classes +

Pizza  
that hasTopping min 5 Thing

Superclasses +

Pizza

Inherited anonymous classes –

hasBase some PizzaBase

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties

Asserted Class Hierarchy: PizzaWithThreeOrLessToppings

Thing  
DomainConcept  
Food  
Pizza  
PizzaWithThreeOrLessToppings  
PizzaWithFiveOrMoreToppings  
PizzaWithTwoToppings  
CheeseyOrVegetarianPizza  
CheeseyPizza  
CheesyAndVegetarianPizza  
MeatyPizza  
NamedPizza  
ThinAndCrispyPizza  
VegetarianPizza  
PizzaBase  
PizzaTopping  
NonVegetarianTopping  
VegetarianTopping  
VegetarianToppingNick  
CheeseTopping  
FishTopping  
FruitTopping  
HerbSpiceTopping  
MeatTopping  
NutTopping  
SauceTopping  
VegetableTopping

Equivalent classes +

Pizza  
that hasTopping max 3 Thing

Class Annotations Class Usage

Class Description: PizzaWithThreeOrLessToppings

Equivalent classes +

Pizza  
that hasTopping max 3 Thing

Superclasses +

Pizza

Inherited anonymous classes –

hasBase some PizzaBase

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Inferred class hierarchy: PizzaWithFiveOrMoreToppings

Annotations

Class Annotations Class Usage

Class Description: PizzaWithFiveOrMoreToppings

Equivalent classes

Pizza  
that hasTopping min 5 Thing

Superclasses

Pizza

Inherited anonymous classes

hasBase some PizzaBase

Instances

Disjoint classes

:-)

Thing

Nothing

DomainConcept

Food

Pizza

CheeseyOrVegetarianPizza

MeatyPizza

NamePizza

PizzaWithFiveOrMoreToppings

AmericanHot

Cajun

Capricciosa

Fiorentina

FourSeasons

Giardiniera

LaReine

Napoletana

Parmense

PolloAdAstra

PrinceCarlo

Siciliana

SloppyGiuseppe

Soho

Veneziana

PizzaWithThreeOrLessToppings

ThinAndCrispyPizza

PizzaBase

PizzaTopping

Asserted class hierarchy Inferred class hierarchy

The screenshot shows the Protege 2007 interface with the ontology file `pizza.owl` loaded. The top menu bar includes File, Edit, Reasoner, Tools, Refactor, Tabs, View, Window, and Help. Below the menu is a toolbar with icons for back, forward, and search, followed by a dropdown menu set to `http://www.co-ode.org/ontologies/pizza/pizza.owl`. The main window has tabs for Active Ontology, Entities, Classes, Object Properties, Data Properties, Individuals, OWLViz, and DL Query. The Classes tab is active.

The left panel displays the Inferred class hierarchy. A red circle highlights the class `PizzaWithThreeOrLessToppings`, which is shown as a subclass of `Pizza`. Below it, `PizzaWithTwoToppings` is also listed as a subclass. Other classes visible include `Thing`, `Nothing`, `DomainConcept`, `Food`, `Pizza`, `CheeseyOrVegetarianPizza`, `MeatyPizza`, `NamedPizza`, `PizzaWithFiveOrMoreToppings`, and `PizzaBase`.

The right panel shows the Class Annotations for `PizzaWithThreeOrLessToppings`. It contains a single annotation: `Pizza that hasTopping max 3 Thing`. The Class Description panel below it provides the same information: `Pizza that hasTopping max 3 Thing`.

**PizzaWithTwoToppings is correctly recognized as a subclass of PizzaWithThreeOrLessToppings...  
... but MargheritaPizza is not recognized as a PizzaWithTwoToppings (hint...)**

Open world  
assumption

# Open VS Closed World Reasoning

- Remember a few slides ago ???
- MargheritaPizza  $\in (\exists \text{ hasTopping Mozzarella}) \sqcap (\exists \text{ hasTopping Tomato})$
- VegetarianPizza = Pizza  $\sqcap (\forall \text{ hasTopping VegetarianTop.})$
- Tomato and Mozzarella ARE Vegetarian toppings
- So, why isn't Margherita classified under VegetarianPizza ?

# Open VS Closed World Reasoning

- Remember a few slides ago ???
- MargheritaPizza  $\in (\exists \text{ hasTopping Mozzarella}) \sqcap (\exists \text{ hasTopping Tomato})$
- VegetarianPizza = Pizza  $\sqcap (\forall \text{ hasTopping VegetarianTop.})$
- Tomato and Mozzarella ARE Vegetarian toppings
- **Because some Margheritas may have other toppings (e.g. HotSpicedBeefTopping) !**

# Open VS Closed World Reasoning

- Closed-World reasoning
  - Negation as failure
  - Anything that cannot be found is false
  - Reasoning about this world
- Open-World reasoning
  - Negation as contradiction
  - Anything might be true unless it can be proven false
  - Reasoning about any world consistent with the model

# Need for closure

Margherita pizzas only have Tomato and Mozzarella for topping

- MargheritaPizza ⊑ ( $\exists \text{ hasTopping}$  Mozzarella)  $\sqcap$  ( $\exists \text{ hasTopping}$  Tomato)  $\sqcap$  ?????

# Need for closure

Margherita pizzas only have Tomato and Mozzarella for topping

- MargheritaPizza ⊑ ( $\exists \text{ hasTopping Mozzarella}$ )  $\sqcap$  ( $\exists \text{ hasTopping Tomato}$ )  $\sqcap$  ( $\forall \text{ hasTopping } ???$ )

# Need for closure

Margherita pizzas only have Tomato and Mozzarella for topping

- MargheritaPizza ⊑ ( $\exists \text{ hasTopping Mozzarella}$ )  $\sqcap$  ( $\exists \text{ hasTopping Tomato}$ )  $\sqcap$  ( $\forall \text{ hasTopping (Mozzarella} \sqcup \text{Tomato)}$ )

# Need for closure

Margherita pizzas only have Tomato and Mozzarella for topping

- MargheritaPizza ⊑ ( $\exists \text{ hasTopping Mozzarella}$ )  $\sqcap$  ( $\exists \text{ hasTopping Tomato}$ )  $\sqcap$  ( $\forall \text{ hasTopping (Mozzarella} \sqcup \text{Tomato)}$ )
- The universal constraint ( $\forall$ ) alone is not enough ! We need both  $\exists$  and  $\forall$  constraints

# Need for closure

Margherita pizzas only have Tomato and Mozzarella for topping

- MargheritaPizza ⊑ ( $\exists \text{ hasTopping Mozzarella}$ )  $\sqcap$  ( $\exists \text{ hasTopping Tomato}$ )  $\sqcap$  ( $\forall \text{ hasTopping (Mozzarella} \sqcup \text{Tomato)}$ )
- Same principle for all the other pizza!

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-02.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data

Asserted Class Hierarchy: Margherita

Superclasses +

- NamedPizza
- hasTopping some MozzarellaTopping
- hasTopping some TomatoTopping
- hasTopping only (MozzarellaTopping or TomatoTopping)

CheeseyPizza

Superclasses +

- NamedPizza
- hasTopping some MozzarellaTopping
- hasTopping some TomatoTopping
- hasTopping only (MozzarellaTopping or TomatoTopping)

CheeseyPizza

Inherited anonymous classes –

- hasBase some PizzaBase
- VegetarianPizza  
or CheeseyPizza
- Pizza  
that hasTopping some CheeseTopping

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/...]

File Edit Reasoner Tools Refactor Tabs View Window

← → http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Properties

Asserted Class Hierarchy: American

Class A  
Annotat

Class D  
Equivale

Superclasses +

ThinAndCrispyPizza

CheeseyOrVegetarianPizza

CheeseyPizza

CheesyAndVegetarianPizza

MeatyPizza

NamedPizza

- American
- AmericanHot
- Cajun
- Capricciosa
- Caprina
- Fiorentina
- FourSeasons
- FruttiDiMare
- Giardiniera
- LaReine
- Margherita
- Mushroom
- Napoletana
- Parmense
- PolloAdAstra
- PrinceCarlo
- QuattroFormaggi
- Rosa
- Siciliana
- SloppyGiuseppe
- Soho
- Veneziana

Inferred class hierarchy

Superclasses +

NamedPizza

hasTopping some MozzarellaTopping

hasTopping some PeperoniSausageTopping

hasTopping some TomatoTopping

hasTopping only MozzarellaTopping  
or TomatoTopping  
or PeperoniSausageTopping)

CheeseyPizza

MeatyPizza

Superclasses +

NamedPizza

hasTopping some MozzarellaTopping

hasTopping some PeperoniSausageTopping

hasTopping some TomatoTopping

hasTopping only (MozzarellaTopping  
or TomatoTopping  
or PeperoniSausageTopping)

CheeseyPizza

MeatyPizza

Inherited anonymous classes –

hasBase some PizzaBase

VegetarianPizza

or CheeseyPizza

http://www.co-ode.org/ontologies/pizza/pizza.owl

File Edit Reasoner Tools Refactor Tabs View Wi

Active Ontology Entities Classes Object Properties D

Asserted Class Hierarchy: Caprina

CheeseyOrVegetarianPizza  
CheeseyPizza  
CheesyAndVegetarianPizza  
MeatyPizza  
NamedPizza  
American  
AmericanHot  
Cajun  
Capricciosa  
Caprina  
Fiorentina  
FourSeasons  
FruttiDiMare  
Giardiniera  
LaReine  
Margherita  
Mushroom  
Napoletana  
Parmense  
PolloAdAstra  
PrinceCarlo  
QuattroFormaggi  
Rosa  
Siciliana  
SloppyGiuseppe  
Soho  
Veneziana  
ThinAndCrispyPizza

Superclasses +

- NamedPizza
- hasTopping some GoatsCheeseTopping
- hasTopping some MozzarellaTopping
- hasTopping some SundriedTomatoTopping
- hasTopping some TomatoTopping
- hasTopping only (MozzarellaTopping  
or TomatoTopping  
or GoatsCheeseTopping  
or SundriedTomatoTopping)

CheeseyPizza

Superclasses +

- NamedPizza
- hasTopping some GoatsCheeseTopping
- hasTopping some MozzarellaTopping
- hasTopping some SundriedTomatoTopping
- hasTopping some TomatoTopping
- hasTopping only (MozzarellaTopping  
or TomatoTopping  
or GoatsCheeseTopping  
or SundriedTomatoTopping)

Inherited anonymous classes –

- hasBase some PizzaBase
- VegetarianPizza

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-04.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties

Inferred class hierarchy: VegetarianPizza

Annotations comment Any pizza

Class Descriptions

Equivalent classes

Pizza  
that hasTopping **only** VegetarianTopping

Superclasses

CheeseyOrVegetarianPizza

Inherited anonymous classes

**hasBase some PizzaBase**

**VegetarianPizza**  
**or** CheeseyPizza

Instances

Disjoint classes

**PIZZA**

**CheeseyOrVegetarianPizza**

**CheeseyPizza**

**VegetarianPizza**

**CheesyAndVegetarianPizza**

**Caprina**

**Margherita**

The screenshot shows the Protege 2007 interface with the ontology file 'pizza.owl' loaded. The left pane displays the 'Inferred class hierarchy' for the class 'VegetarianPizza'. A red arrow points from the 'VegetarianPizza' node in the tree to its detailed description in the right pane. The right pane shows the following details:

- Annotations:** comment Any pizza
- Class Descriptions:** Pizza that hasTopping **only** VegetarianTopping
- Superclasses:** CheeseyOrVegetarianPizza
- Inherited anonymous classes:** **hasBase some PizzaBase**, **VegetarianPizza** **or** CheeseyPizza
- Instances:** Caprina, Margherita
- Disjoint classes:** (empty)

The inferred class hierarchy tree on the left includes nodes for Thing, Nothing, DomainConcept, Food, Pizza, CheeseyOrVegetarianPizza, CheeseyPizza, VegetarianPizza, CheesyAndVegetarianPizza, Caprina, Margherita, MeatyPizza, NamedPizza, PizzaWithFiveOrMoreToppings, PizzaWithThreeOrLessToppings, ThinAndCrispyPizza, PizzaBase, and PizzaTopping.

# Getting in sync!

If you need to catch-up,  
the ontology at this point is  
protege2007owlTutorial-03.owl

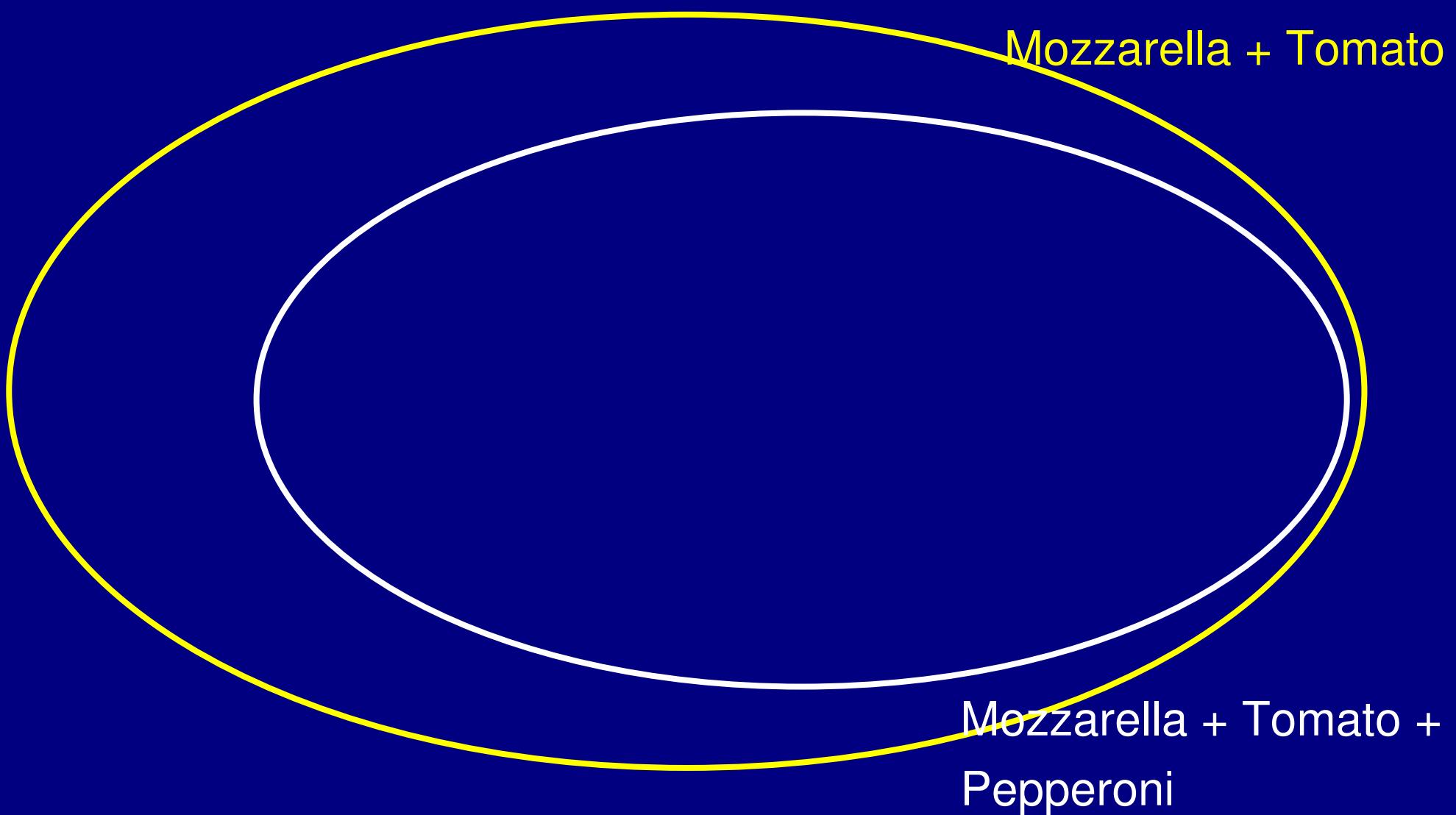
from:

<http://www.ea3888.univ-rennes1.fr/dameron/protege2007/>

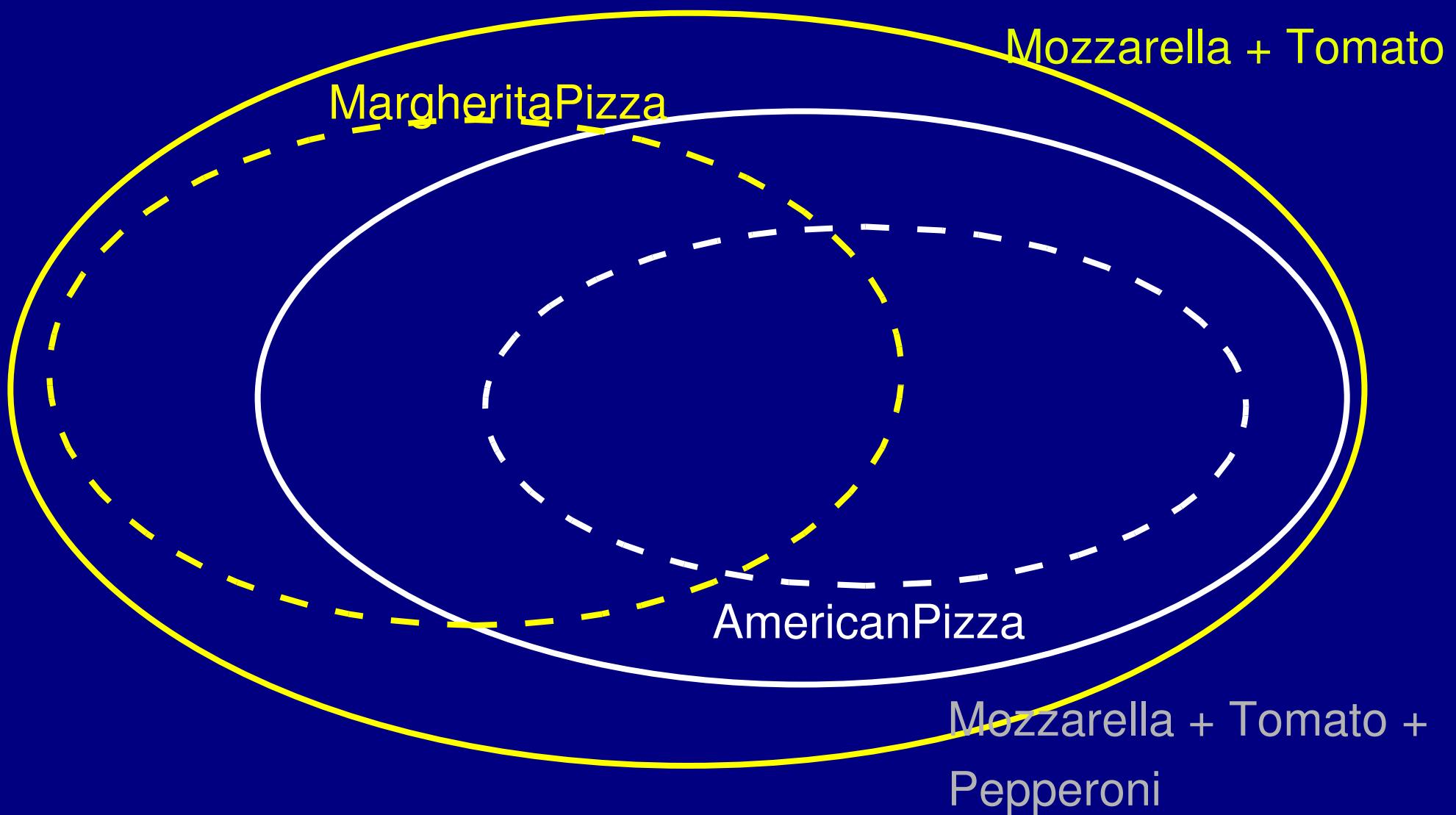
# More fun with closure and defined classes

- Before we added the closures, why wasn't AmericanPizza recognised as a subclass of MargheritaPizza ?

# Need for closure



# Need for closure



# Need for closure

Margherita pizzas only have Tomato and Mozzarella for topping

- MargheritaPizza = ( $\exists \text{ hasTopping}$  Mozzarella)  $\sqcap$  ( $\exists \text{ hasTopping}$  Tomato)  $\sqcap$  ( $\forall \text{ hasTopping}$  (Mozzarella  $\sqcup$  Tomato))

# Getting in sync!

If you need to catch-up,  
the ontology at this point is  
protege2007owlTutorial-04.owl

from:

<http://www.ea3888.univ-rennes1.fr/dameron/protege2007/>

# More fun with cardinality

- Why isn't MargheritaPizza classified under PizzaWithTwoToppings ?

# More fun with cardinality

- Why isn't MargheritaPizza classified under PizzaWithTwoToppings?
- Hints:
  - Why isn't it classified under PizzaWithThreeOrLessToppings ?
  - Why isn't it even classified under PizzaWithFiveOrMoreToppings ?... do Margherita pizza have exactly 4 toppings ?

# More fun with cardinality

- Why isn't MargheritaPizza classified under PizzaWithTwoToppings?
- Still... the open-world assumption:  
imagine one instance of MargheritaPizza  
having as topping:
  - one instance of MozzarellaTopping
  - one other instance of MozzarellaTopping
  - one instance of TomatoTopping

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: Margherita2

Annotations +

Class Annotations Class Usage

Class Description: Margherita2

Equivalent classes +

hasTopping some MozzarellaTopping  
and hasTopping some TomatoTopping  
and hasTopping exactly 2 Thing

Superclasses +

NamedPizza

Inherited anonymous classes –

hasBase some PizzaBase

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Properties

Inferred class hierarchy: Margherita2

Annotations

Annotations +

Class Annotations Class Usage

Class Description: Margherita2

Equivalent classes +

hasTopping some MozzarellaTopping  
and hasTopping some TomatoTopping  
and hasTopping exactly 2 Thing

Superclasses +

NamedPizza  
CheesyAndVegetarianPizza  
PizzaWithTwoToppings

Inherited anonymous classes –

hasBase some PizzaBase  
Pizza  
that hasTopping exactly 2 Thing  
Pizza  
that hasTopping max 3 Thing  
VegetarianPizza

Asserted class hierarchy Inferred class hierarchy

**Superclasses**

**NamedPizza**

**CheesyAndVegetarianPizza**

**PizzaWithTwoToppings**

The screenshot shows the Protege 2007 OWL tutorial interface. The main window displays the inferred class hierarchy for the ontology pizza.owl. The class Margherita2 is selected, highlighted with a blue border. The right-hand panel is focused on the 'Superclasses' tab, which lists three superclasses: NamedPizza, CheesyAndVegetarianPizza, and PizzaWithTwoToppings. A red box highlights this list. Below the superclasses, the 'Class Description' tab is open, showing the asserted class description for Margherita2: it hasTopping some MozzarellaTopping, hasTopping some TomatoTopping, and hasTopping exactly 2 Thing. The bottom section shows inherited anonymous classes, including PizzaBase, Pizza, and VegetarianPizza.



# hasValue restriction

- So far, we have been narrowing the range of relationship
  - create the class Person
  - create the relation *hasPizzaMaker*: Pizza -> Person
  - create ItalianPerson as subclass of Person
  - define GenuinePizza = ( $\exists$  *hasPizzaMaker* ItalianPers.)

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: Person

Annotations

Class Annotations: Person

Class Description: Person

Disjoint classes

Food

Annotations

Class Annotations Class Usage

Equivalent classes

Superclasses

DomainConcept

Inherited anonymous classes

Instances

Disjoint classes

Food

Asserted class hierarchy Inferred class hierarchy

The screenshot shows the Protege 2007 interface with the ontology file `pizza.owl` loaded. The main window displays the asserted class hierarchy for the class `Person`. The hierarchy tree on the left shows the following structure:

- `Thing`
- `DomainConcept`
- `Person` (selected)
- `FrenchPerson`
- `ItalianPerson`
- `Food`
- `Pizza`
- `PizzaBase`
- `PizzaTopping`

The `Person` class is highlighted with a blue selection bar. In the bottom-left corner of the main window, there are two tabs: `Asserted class hierarchy` and `Inferred class hierarchy`, with `Asserted class hierarchy` currently selected.

On the right side of the interface, there are three panels for the `Person` class:

- Annotations:** An empty panel.
- Class Annotations:** An empty panel.
- Class Description:** This panel contains the following sections:
  - Equivalent classes:** An empty list.
  - Superclasses:** A list containing `DomainConcept`.
  - Inherited anonymous classes:** An empty list.
  - Instances:** An empty list.
  - Disjoint classes:** A list containing `Food`. The entry for `Food` is circled in red.

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Object Properties: hasPizzaMaker      Annotations: hasPizzaMaker

Annotations +

Annotations Object Property Usage

Character      Description: hasPizzaMaker      Domains and ranges: hasPizzaM

Functional

Inverse functi...

Transitive

Symmetric

Antisymmetric

Reflexive

Irreflexive

Equivalent object properties +

Super properties +

Inverse properties +

Disjoint properties +

Property chains +

Domains (intersection) +

Pizza

Ranges (intersection) +

Person

The screenshot displays the Protege 2007 interface for editing an ontology. The main window shows the 'Active Ontology' tab selected, displaying the 'hasPizzaMaker' object property. This property is defined with domains (Intersection of Pizza and Person) and ranges (Intersection of Person and Pizza). The 'Character' tab indicates that 'Functional' is checked. The 'Description' tab lists various property relationships: Equivalent object properties, Super properties, Inverse properties, Disjoint properties, and Property chains. The 'Annotations' tab is currently empty. The bottom tabs provide navigation between different aspects of the property definition.

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties

Asserted Class Hierarchy: GenuinItalianPizza

Equivalent classes +

Pizza  
that hasPizzaMaker some ItalianPerson

Class Annotations Class Usage

Class Description: GenuinItalianPizza

Equivalent classes +

Pizza  
that hasPizzaMaker some ItalianPerson

Superclasses +

Inherited anonymous classes –

hasBase some PizzaBase

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

The screenshot displays the Protege 2007 interface for editing an ontology. On the left, the 'Asserted Class Hierarchy' panel shows a tree structure of classes under 'GenuinItalianPizza'. The tree starts with 'Thing', then branches into 'DomainConcept', 'Person' (with subclasses 'FrenchPerson' and 'ItalianPerson'), 'Food', and 'Pizza' (with subclasses like 'GenuinItalianPizza', 'CheeseyOrVegetarianPizza', etc.). On the right, the main workspace is divided into several tabs for managing the 'GenuinItalianPizza' class. The 'Equivalent classes' tab is currently active, containing the asserted rule 'that hasPizzaMaker some ItalianPerson'. Below it, the 'Class Description' tab also displays this rule. Other tabs include 'Superclasses', 'Inherited anonymous classes', 'Instances', and 'Disjoint classes'. The entire 'Equivalent classes' section is highlighted with a red box.

# hasValue restriction

- So far, we have been narrowing the range of relationship
- We may also want to restrict it to a precise value (and not to a set of values)
  - create olivier as an instance of Person
  - define OliviersPizza = (*hasPizzaMaker*  $\ni$  olivier)

# hasValue restriction

- Create luigi as an instance of ItalianPerson
- Create LuigisPizza
- Classify :-)

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Asserted Class Hierarchy: ItalianPerson      Class Annotations: ItalianPerson

Annotations +

Class Annotations Class Usage

Class Description: ItalianPerson

Equivalent classes +

Superclasses +

Person

Inherited anonymous classes -

Instances +

luigi

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

The screenshot shows the Protege 2007 interface with the ontology file `pizza.owl` loaded. The main window displays the asserted class hierarchy on the left and various annotations for the `ItalianPerson` class on the right.

**Asserted Class Hierarchy:** Shows the inheritance path from `Thing` through `DomainConcept`, `Person`, `FrenchPerson`, and finally `ItalianPerson`. Other food-related classes like `Pizza`, `PizzaBase`, and `PizzaTopping` are also listed.

**Class Annotations:** The `ItalianPerson` class has no annotations listed.

**Class Description:** The `ItalianPerson` class is described as having the same equivalent classes as `Person`. It has one instance named `luigi`.

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies/pizza/pi

Active Ontology Entities Classes Object Properties

Asserted Class Hierarchy: LuigisPizza

Thing  
DomainConcept  
Person  
FrenchPerson  
ItalianPerson  
Food  
Pizza  
LuigisPizza  
GenuineItalianPizza  
CheeseyOrVegetarianPizza  
CheeseyPizza  
CheesyAndVegetarianPizza  
MeatyPizza  
NamedPizza  
PizzaWithFiveOrMoreToppings  
PizzaWithThreeOrLessToppings  
PizzaWithTwoToppings  
ThinAndCrispyPizza  
VegetarianPizza  
PizzaBase  
PizzaTopping

Equivalent classes +

Pizza  
that hasPizzaMaker value luigi

Class Annotations Class Usage

Class Description: LuigisPizza

Equivalent classes +  
Pizza  
that hasPizzaMaker value luigi

Superclasses +

Inherited anonymous classes –

hasBase some PizzaBase

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Pro

Inferred class hierarchy: LuigisPizza

Class Annotations Class Usage

Class Description: LuigisPizza

Equivalent classes +

Pizza  
that hasPizzaMaker value luigi

Superclasses +

GenuinItalianPizza

Inherited anonymous classes –

hasBase some PizzaBase

Instances +

Disjoint classes +

Thing

Nothing

DomainConcept

Food

Pizza

CheeseyOrVegetarianPizza

GenuinItalianPizza

LuigisPizza

MeatyPizza

NamedPizza

PizzaWithFiveOrMoreToppings

PizzaWithThreeOrLessToppings

ThinAndCrispyPizza

PizzaBase

PizzaTopping

Person

**GenuinItalianPizza**

**LuigisPizza**

**MeatyPizza**

**NamedPizza**

Inferred class hierarchy

# (slightly off topic) remark

- ItalianPerson and FrenchPerson are not disjoint
- Because there is no Unique Name Assumption, luigi and olivier could be the same person
- Use the owl:differentFrom and owl:allDifferent constructs (in the OWL menu)!



# Reasoning makes life easier :-)

- Supports queries such as:
  - What are the vegetarian pizza ?
  - What are the cheesy pizza ?
  - What are the non-cheesy pizza ?
  - What are the cheesy vegetarian pizza ?
- ... it allows you to take advantage of the knowledge you put into your ontology



OWL and beyond...  
OWL 1.1

# **Qualified Cardinality Restriction**

- OWL 1.0 Cardinality restrictions:
  - PizzaWithTwoToppings
  - PizzaWithFiveOrMoreToppings
  - PizzaWithThreeToppingsOrLess
- OWL 1.1 Qualified cardinality restrictions
  - PizzaWithThreeCheese
  - PizzaWithAtLeastTwoCheese
  - PizzaWithAtLeastTwoCheeseAnd

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies

Active Ontology Entities Classes Object Properties

Asserted Class Hierarchy: ThreeCheesePizza

Thing  
DomainConcept  
Person  
Food  
Pizza  
ThreeCheesePizza  
LuigisPizza  
GenuineItalianPizza  
CheeseyOrVegetarianPizza  
CheeseyPizza  
CheesyAndVegetarianPizza  
MeatyPizza  
NamedPizza  
PizzaWithFiveOrMoreToppings  
PizzaWithThreeOrLessToppings  
PizzaWithTwoToppings  
ThinAndCrispyPizza  
VegetarianPizza  
PizzaBase  
PizzaTopping

Equivalent classes +

Pizza  
that hasTopping exactly 3 CheeseTopping

Class Annotations Class Usage

Class Description: ThreeCheesePizza

Equivalent classes +

Pizza  
that hasTopping exactly 3 CheeseTopping

Superclasses +

Inherited anonymous classes –

hasBase some PizzaBase

Instances +

Disjoint classes +

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties

Asserted Class Hierarchy: PizzaWithAtLeastTwoCheese

Equivalent classes +

Pizza  
that hasTopping min 2 CheeseTopping

Class Annotations Class Usage

Class Description: PizzaWithAtLeastTwoCheese

Equivalent classes +

Pizza  
that hasTopping min 2 CheeseTopping

Superclasses +

CheeseyPizza

Inherited anonymous classes –

hasBase some PizzaBase

VegetarianPizza  
or CheeseyPizza

Pizza  
that hasTopping some CheeseTopping

Instances +

Disjoint classes +

Thing

DomainConcept

Person

Food

Pizza

- PizzaWithAtLeastTwoCheese
- ThreeCheesePizza
- LuigisPizza
- GenuineItalianPizza
- CheeseyOrVegetarianPizza
- CheeseyPizza
- CheesyAndVegetarianPizza
- MeatyPizza
- NamedPizza
- PizzaWithFiveOrMoreToppings
- PizzaWithThreeOrLessToppings
- PizzaWithTwoToppings
- ThinAndCrispyPizza
- VegetarianPizza

PizzaBase

PizzaTopping

Asserted class hierarchy Inferred class hierarchy

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties

Inferred class hierarchy: PizzaWithAtLeastTwoCheese

Annotations

Class Description

Equivalent classes

Pizza  
that hasTopping min 2 CheeseTopping

Superclasses

CheeseyPizza

Inherited anonymous classes

hasBase some PizzaBase

VegetarianPizza  
or CheeseyPizza

Pizza  
that hasTopping some CheeseTopping

Instances

Disjoint classes

Thing

- Nothing
- DomainConcept
- Food
  - Pizza
    - CheeseyOrVegetarianPizza
      - CheeseyPizza
        - American
        - AmericanHot
        - Cajun
        - Capricciosa
      - CheesyAndVegetarianPizza
      - FourSeasons
      - Giardiniera
      - LaReine
      - Mushroom
      - Napoletana
    - PizzaWithAtLeastTwoCheese
      - Caprina
      - Fiorentina
      - Parmense
      - PrinceCarlo
      - Rosa
      - Soho
      - ThreeCheesePizza
      - PolloAdAstra
      - QuattroFormaggi
      - Spicy

# Additional features for properties

- Reflexivity

$$\forall a \in X, aRa$$

- e.g. *knows*, *isGreaterOrEqualTo*

- Irreflexivity

$$\forall a \in X, \neg(aRa)$$

- e.g. *isMotherOf*, *isGreaterThan*

- Antisymmetry

$$\forall a, b \in X, aRb \wedge bRa \Rightarrow a = b$$

- e.g. *isAncestorOf*, *isGreaterOrEqualTo*

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/...]

File Edit Reasoner Tools Refactor Tabs View Window Help

← → http://www.co-ode.org/ontologies/pizza/pizza.owl

Active Ontology Entities Classes Object Properties Data Properties

Object Properties: hasPizzaMaker

hasPizzaMaker  
hasIngredient  
isIngredientOf

### Character

- Functional
- Inverse functi...
- Transitive
- Symmetric
- Antisymmetric
- Reflexive
- Irreflexive

### Description: hasPizzaMaker

Equivalent object properties

Super properties

Inverse properties

Disjoint properties

Property chains

Ranges (intersection)

Person

# Property chains

- Allow to describe (simple) composition of relations
- e.g.:  
if X *eats* Y and Y *hasIngredient* Z  
then X *eats* Z
- Notation:  $f \circ g(x) = f(g(x))$

http://www.co-ode.org/ontologies/pizza/pizza.owl - [/home/olivier/articles/2007/protege/tutorial/protege2007owlTutorial-05.owl]

File Edit Reasoner Tools Refactor Tabs View Window Help

Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query

Object Properties: eats Annotations: eats

eats hasIngredient hasPizzaMaker isEatenBy isIngredientOf

Annotations +

Property chains +

hasIngredient o eats → eats

Functional  
Inverse function...  
Transitive  
Symmetric  
Antisymmetric  
Reflexive  
Irreflexive

Equivalent object properties  
Super properties +  
Inverse properties +  
isEatenBy  
Disjoint properties +  
Property chains +

Domains (intersection)  
Person  
Ranges (intersection)  
Food

A screenshot of the Protege 2007 ontology editor interface. The main window shows the 'Annotations: eats' tab. A property chain 'hasIngredient o eats → eats' is highlighted with a red box and a red arrow points from it towards the bottom right. The left sidebar lists object properties: eats, hasIngredient, hasPizzaMaker, isEatenBy, and isIngredientOf. The right sidebar shows domains and ranges for Person and Food. The bottom left sidebar contains various property checkmarks like Functional, Inverse function..., Transitive, etc.

# Summary

# Summary

1. Compositional approach
2. Intensional description
3. Reasoning
  - classification
  - open-world assumption
  - inconsistency

