Terminology & Ontology in the Digital Age

Renmin University (China)

26-30 October 2020

Monday 26 October	14:00-17:00 pm
Tuesday 27 October	14:00-17:00 pm
Wednesday 28 October	19:00-22:00 pm
Thursday 29 October	**19:00-22:00 pm**
Friday 30 October	19:00-22:00 pm

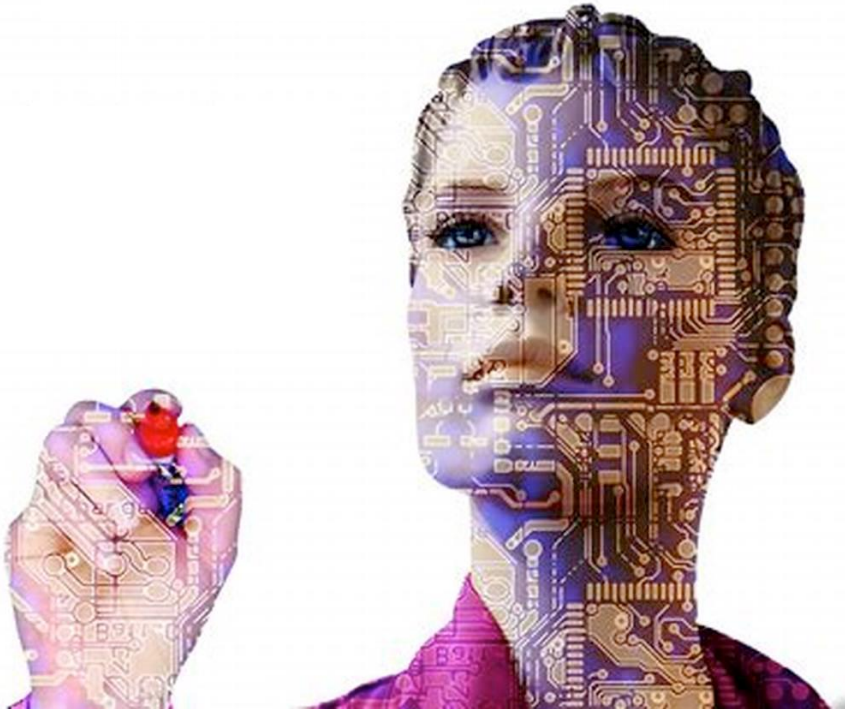
Beijing Time

Prof. Christophe Roche

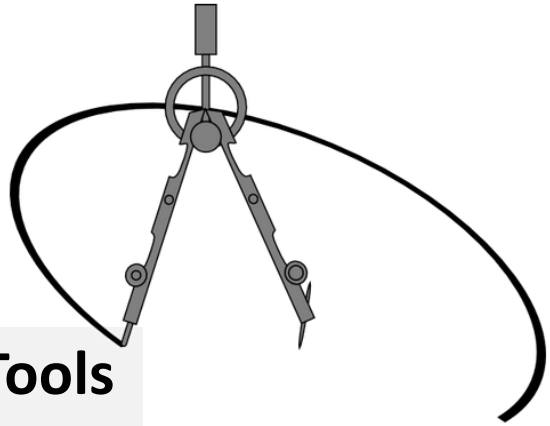
Liaocheng University (China)
Savoie Mont-Blanc University (France)

http://christophe-roche.fr/

Contents



Tools & Environments



- **Graphical Tools:** - CmapTools
- **Formal Tool:** - Protégé
- **Dedicated Tools:** - Tedi

Downloads

CmapTools

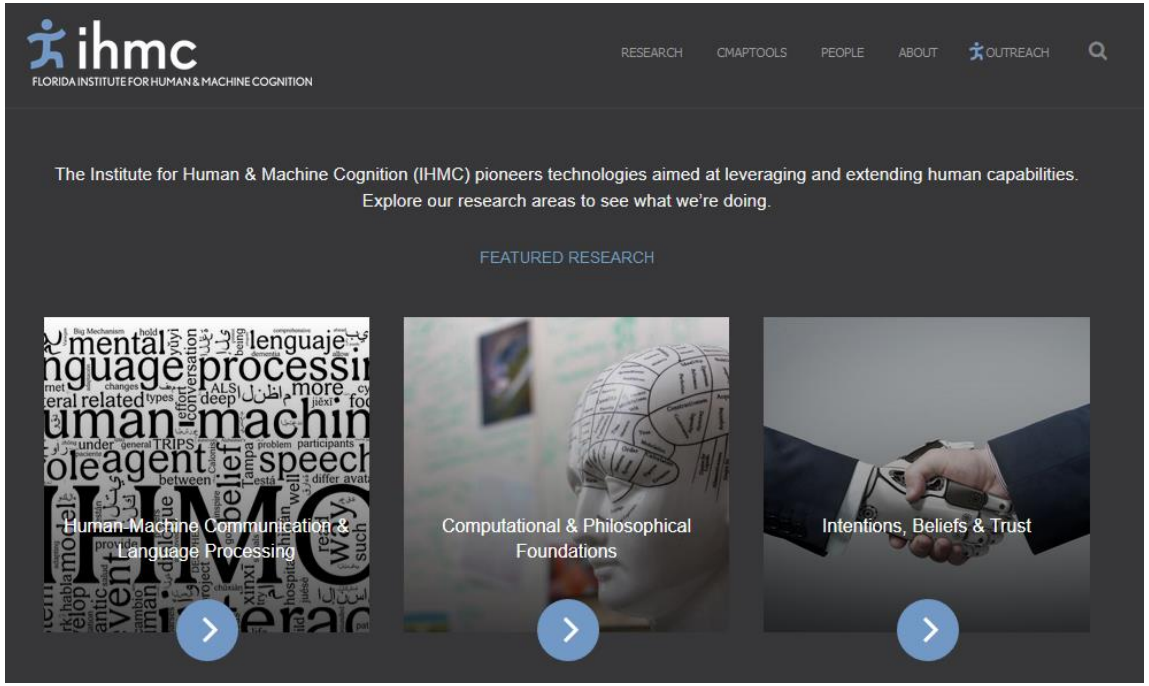
http://cmap.ihmc.us/

Protégé

https://protege.stanford.edu/

CmapTools

http://cmap.ihmc.us/



Concept maps are graphical tools for organizing and representing knowledge.

They include **concepts**, usually enclosed in circles or boxes of some type, and **relationships** between concepts indicated by a connecting line linking two concepts.

☑ Epistemological Principles

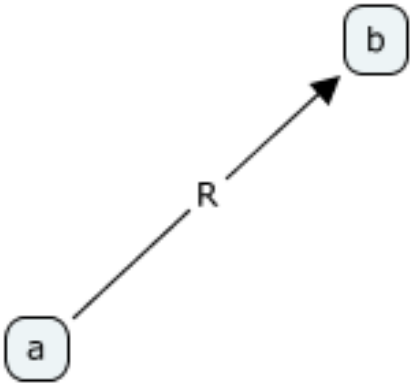
Concept: *Concept* as *a perceived regularity in events or objects, or records of events or objects, designated by a label.*

Proposition: *Propositions* are *statements about some object or event in the universe, either naturally occurring or constructed. Propositions contain two or more concepts connected using linking words or phrases to form a meaningful statement.*

☑ Representation

Concept: *Node*

Proposition: *labeled link*



☑ Epistemological Principles

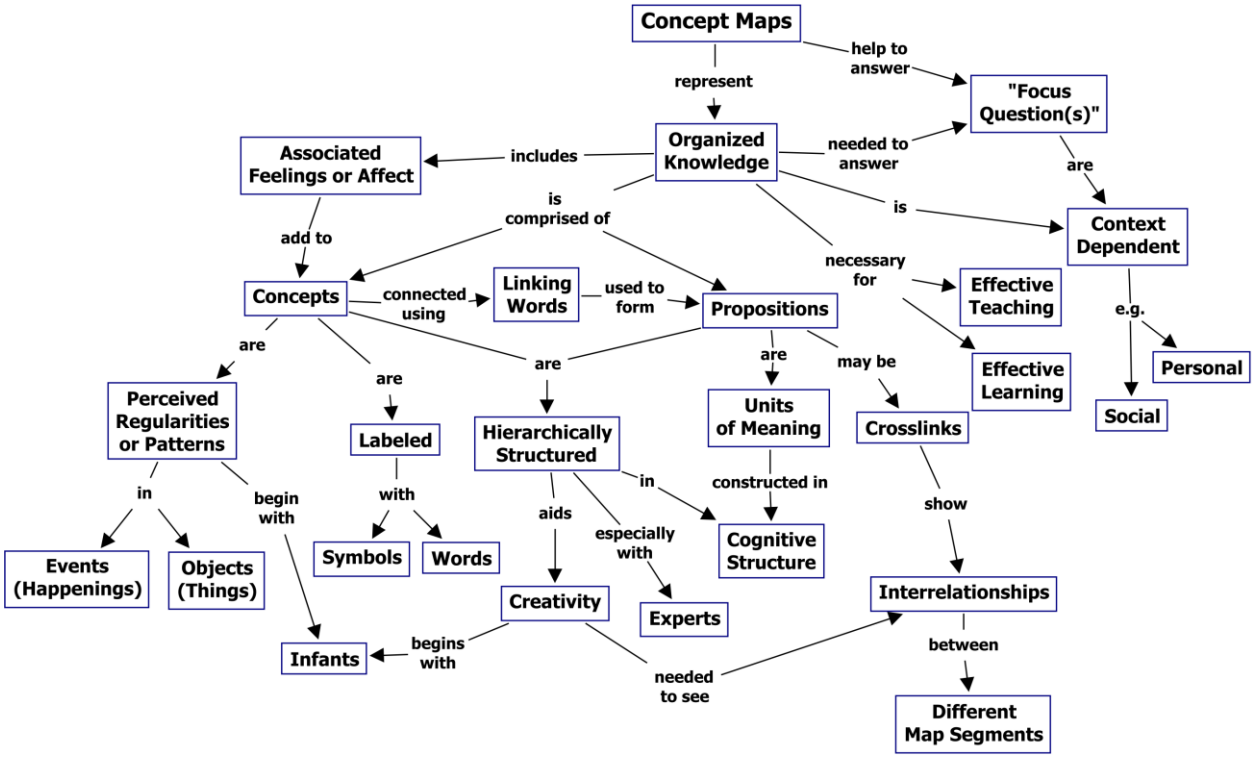
Concept: *Concept as a perceived regularity in events or objects, or records of events or objects, designated by a label.*

Proposition: *Propositions are statements about some object or event in the universe, either naturally occurring or constructed. Propositions contain two or more concepts connected using linking words or phrases to form a meaningful statement.*

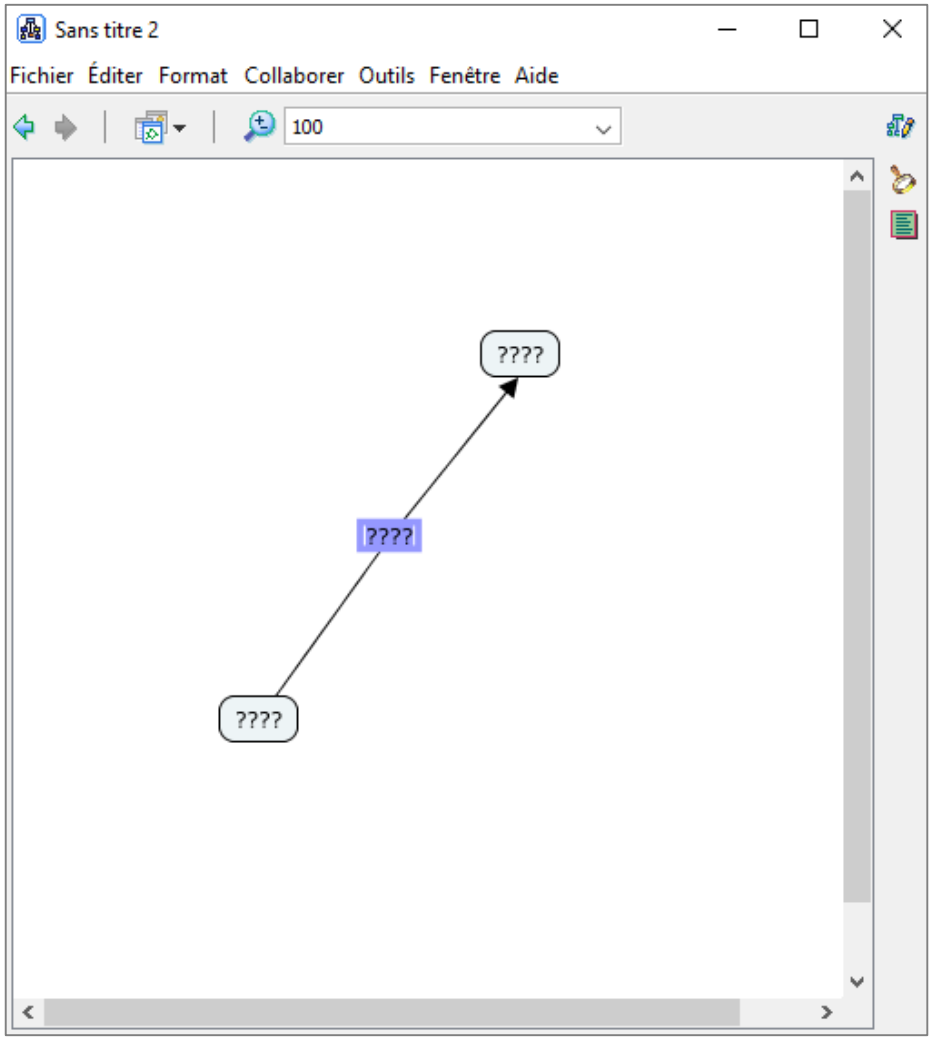
☑ Representation

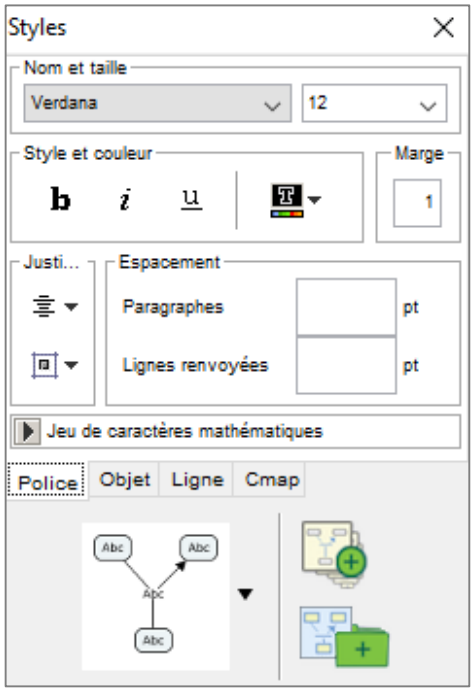
Concept: *Node*

Proposition: *labeled link*



CmapTools





Do it

How to Represent the Categories of Thought?

Langue of Intellection

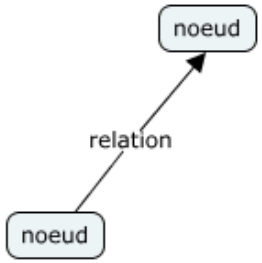
Categories of Thought :

- **Object**
- **Characteristic** :
  - essential
  - descriptive
- **Concept**
- **Class**
- **Relation** :
  - generic
  - partitive
  - functonial
  - associative
  - …….

Representation Language

Language Categories of CmapTools

- **Node**
- **Labelled link** (binary relationhips)



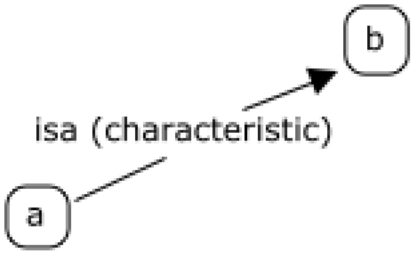
Semiotic code

each category of thought will be represented differently

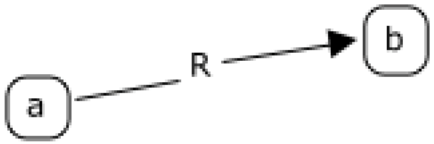
concept — <Concept> — <Concept>

individual — *individual* — *individual*

Essential characteristic — /characteristic/

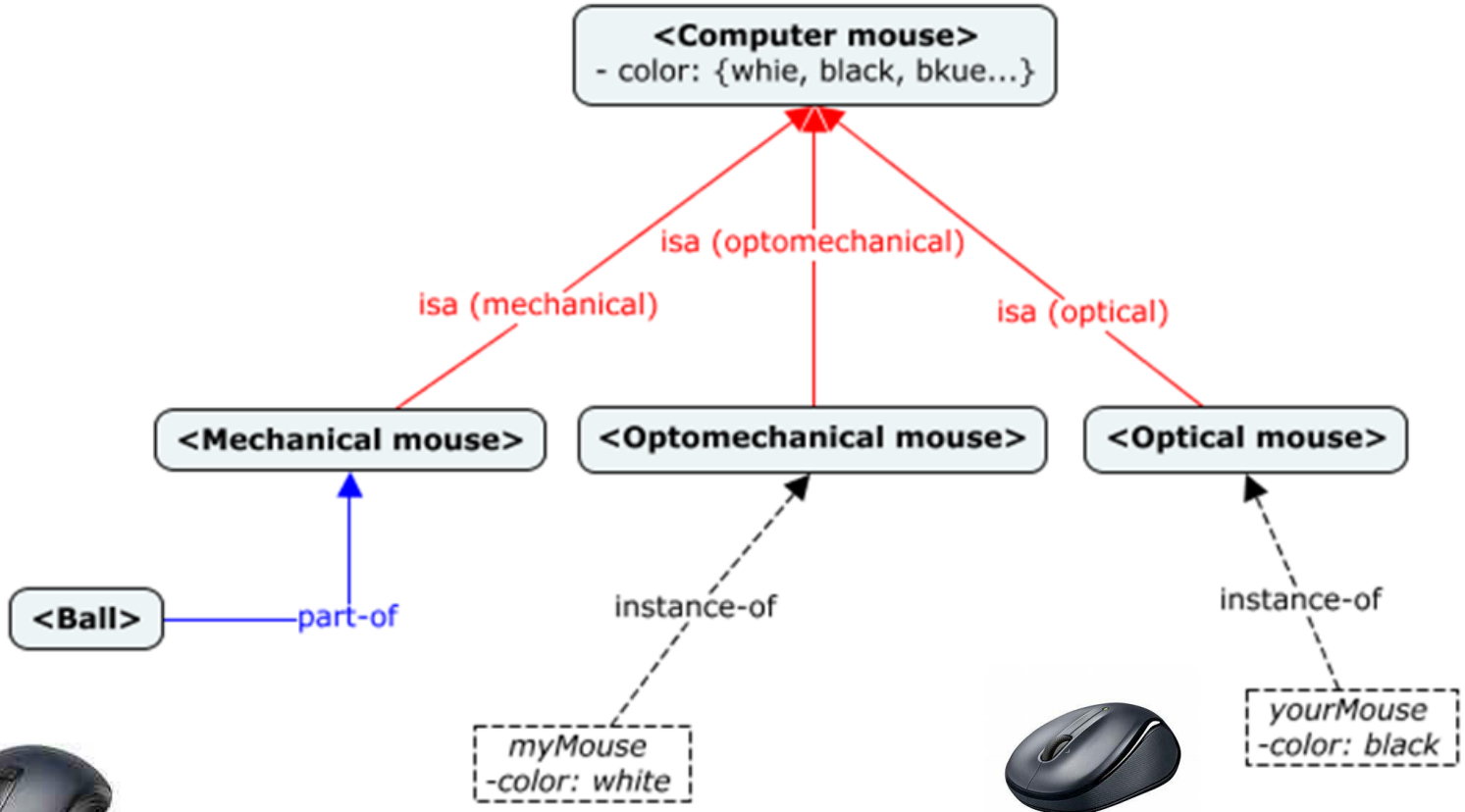


relations — R(a,b)



Descriptive characteristic — at(e,v)

CmapTools



No verification

Exercise

椅子

armchair

chaise

fauteuil

pouf

banc

stool

tabouret

长凳

bench

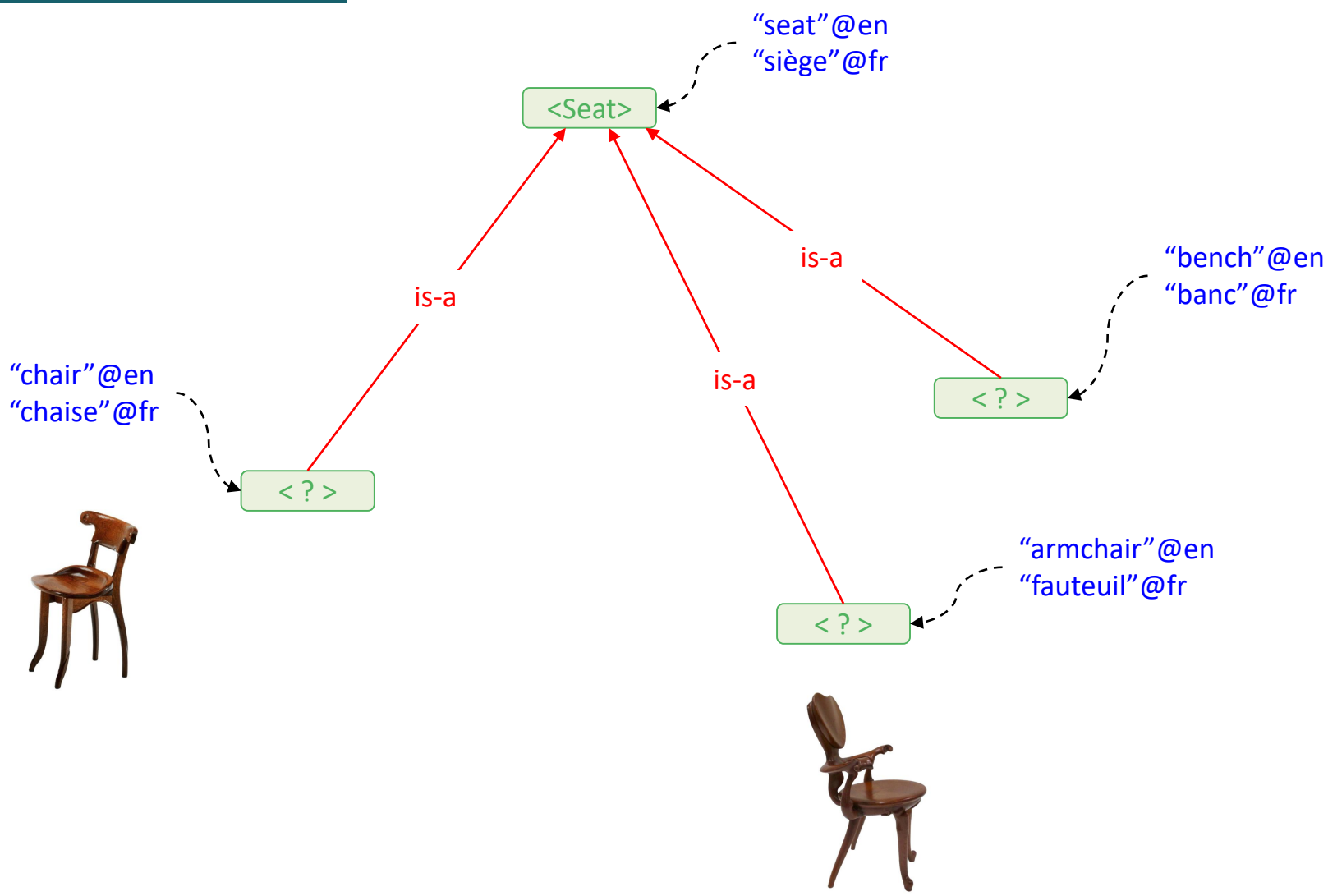
chair

canapé

couch

长椅

CmapTools

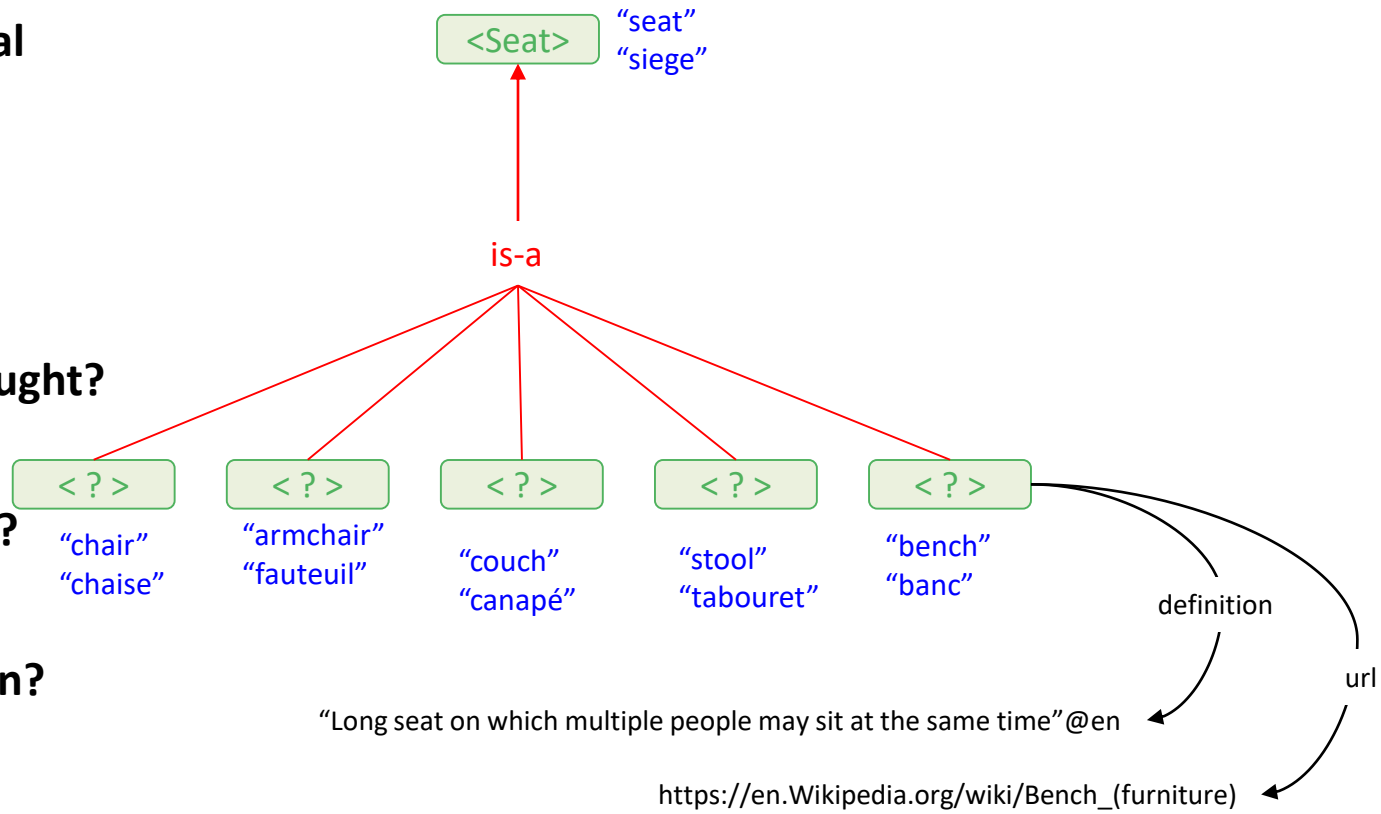

“seat”@en
“siège”@fr
<Seat>
is-a
is-a
is-a
“bench”@en
“banc”@fr
“chair”@en
“chaise”@fr
< ? >
< ? >
< ? >
“armchair”@en
“fauteuil”@fr

CmapTools

✓ **Easy to use**

✓ **Human Readable**

✓ **Semi-Formal**



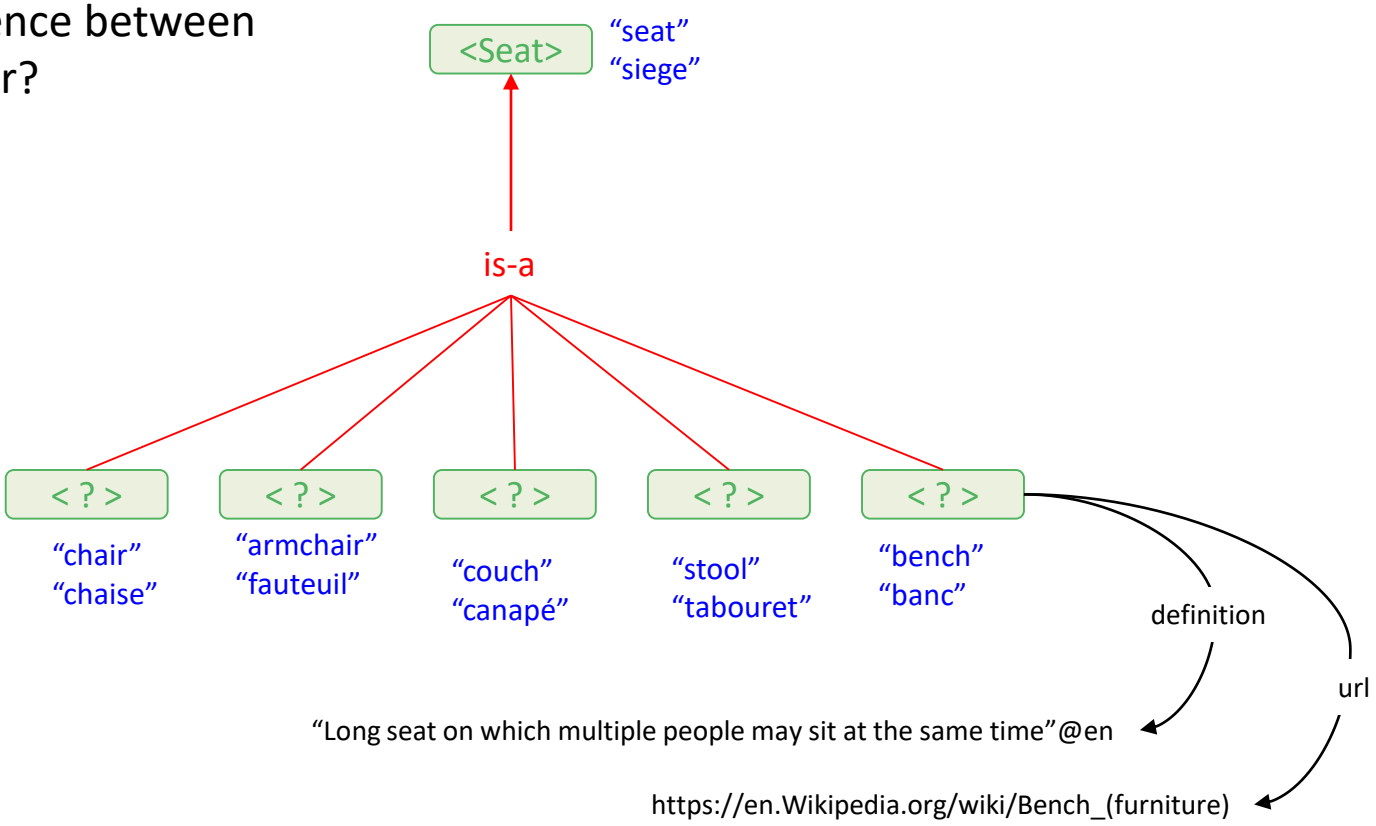
- **Categories of thought?**
- **Methodology?**
- **Formal definition?**
- **Coherency?**
- **Operationalization?**

CmapTools

(Formal) Definition?

Is defining Chair as a subclass of Seat sufficient?

What is the difference between Chair and Armchair?



Array of differences

Objects	Concepts	Axis of analysis		Axis of analysis		Axis of analysis		Axis of analysis		Terms	
Objects	Concepts	for one person	several persons	with feet	without feet	with back	without back	with arms	without arms	Designations (English)	Designations (French)
	<Seat 1 person with feet with back without arms>	X		X		X			X	"chair"	"chaise"
	<Seat 1 person with feet with back with arms>	X		X		X		X		"armchair"	"fauteuil"
	<Seat 1 person with feet without back without arms>	X		X			X		X	"stool"	"tabouret"
	<Seat several persons with feet with back with arms>		X	X		X		X		"couch"	"canapé"
	<Seat several persons with feet without back without arms>		X	X			X		X	"bench"	"banc"

"chair" : Seat for one person with feet and back without arms.

<Seat for one person with feet with back without arms>

::= <Seat> + /for one person/ + /with feet/ + /with back/ + /without arms/

"armchair" : Seat for one person with feet and back with arms.

<Seat for one person with feet with back without arms>

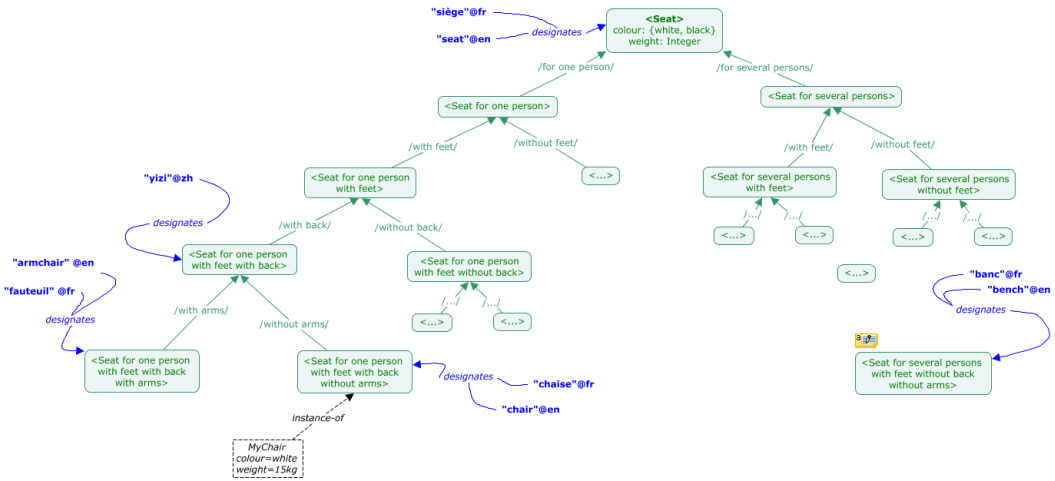
::= <Seat> + /for one person/ + /with feet/ + /with back/ + /with arms/

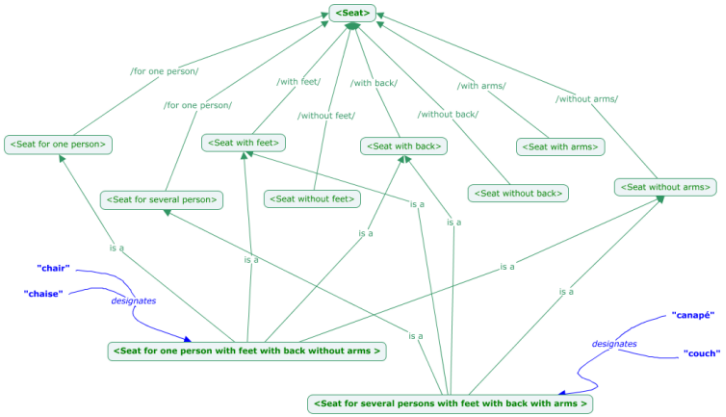
"bench" : Seat for several persons with feet, without back, and without arms.

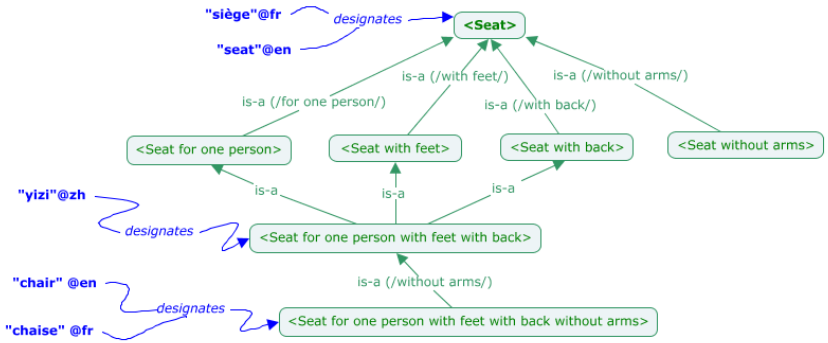
<Seat for one person with feet with back without arms>

::= <Seat> + /for one person/ + /with feet/ + /with back/ + /with arms/

Graphical Notation







They are all equivalent.

A given graph is one of several possible representations of the same conceptual system

<Concept> = { essential characteristics}

protégé

https://protege.stanford.edu/

Protégé is a free, open source ontology editor written in Java and developed at Stanford University.

More than 300,000 users are registered.

WHY PROTÉGÉ

Protégé's plug-in architecture can be adapted to build both simple and complex ontology-based applications. Developers can integrate the output of Protégé with rule systems or other problem solvers to construct a wide range of intelligent systems. Most important, the Stanford team and the vast Protégé community are here to help.

ACTIVE COMMUNITY

Protégé is actively supported by a strong community of users and developers that field questions, write documentation, and contribute plug-ins.

W3C STANDARDS SUPPORT

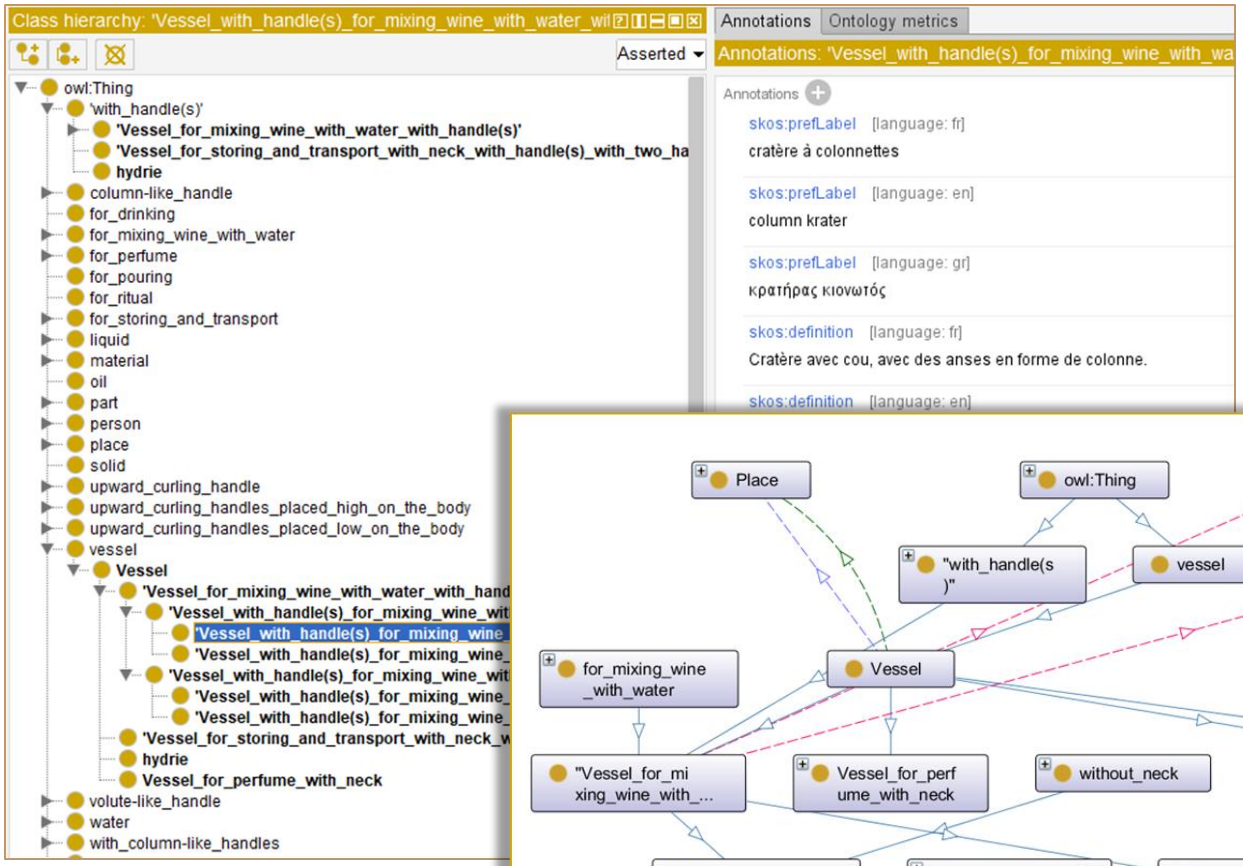
Protégé fully supports the latest OWL 2 Web Ontology Language and RDF specifications from the World Wide Web Consortium.

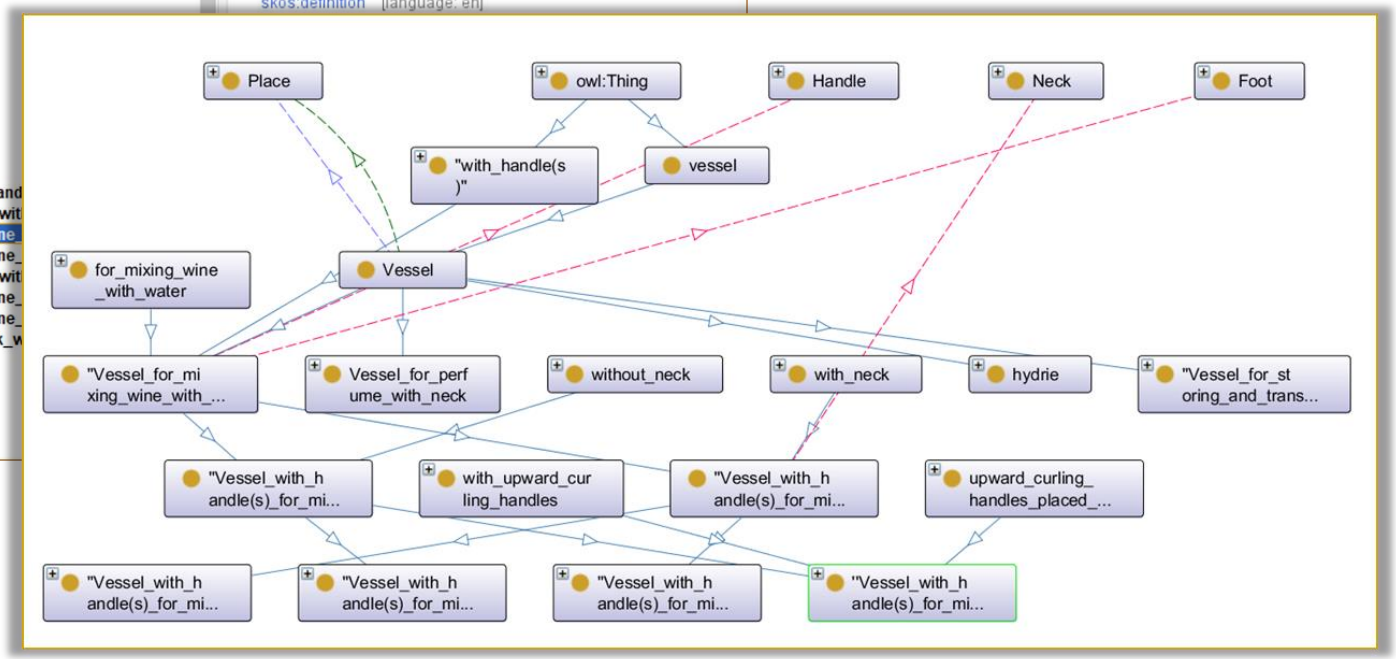
EXTENSIBLE OPEN SOURCE ENVIRONMENT

Protégé is based on Java, is extensible, and provides a plug-and-play environment that makes it a flexible base for rapid prototyping and application development.

Protégé provides:

- a graphic user interface to define ontologies;
- deductive classifiers to validate that models are consistent and to infer new information based on the analysis of ontology.





Protégé https://protege.stanford.edu/

DOWNLOAD NOW

Protégé Desktop is a feature rich ontology editing environment with full support for the OWL 2 Web Ontology Language, and direct in-memory connections to description logic reasoners like HermiT and Pellet.

Protégé Desktop supports creation and editing of one or more ontologies in a single workspace via a completely customizable user interface. Visualization tools allow for interactive navigation of ontology relationships. Advanced explanation support aids in tracking down inconsistencies. Refactor operations available including ontology merging, moving axioms between ontologies, rename of multiple entities, and more.

Screenshots Documentation + Resources

- ✓ *W3C standards compliant*
- ✓ *Customizable user interface*
- ✓ *Visualization support*
- ✓ *Ontology refactoring support*
- ✓ *Direct interface to reasoners*
- ✓ *Highly pluggable architecture*
- ✓ *Cross compatible with WebProtégé*

Download for Windows
Protégé Desktop v.5.5.0

Download platform independent version
(requires a Java Runtime Environment)

Protégé

https://protege.stanford.edu/support.php

DOCUMENTATION

Learn about the Protégé toolset and developing ontologies using our product documentation.

USERS | DEVELOPERS

Look here first!

» *Protégé Frequently Asked Questions*
Find answers to commonly asked questions about the Protégé Project.

» *WebProtégé Overview*
Get an overview of WebProtégé features and the latest project activity.

» *WebProtégé User's Guide*
A short user's guide with screencasts and screenshots describing the main user interface elements in WebProtégé.

» *Protégé Desktop*
Top-level entry point for all documentation pertaining to the latest version of Protégé Desktop, including an overview of editor features, links to tutorials etc.

» *Ontology Development 101*
A guide to creating your first ontology.

» *Protégé OWL Tutorial*
A substantial guide to the Web Ontology Language (OWL) and ontology engineering.

Ontology Development 101: A Guide to Creating Your First Ontology

Natalya F. Noy and Deborah L. McGuinness
Stanford University, Stanford, CA, 94305
noy@smi.stanford.edu and dlm@ksl.stanford.edu

1 Why develop an ontology?

In recent years the development of ontologies—explicit formal specifications of the terms in the domain and relations among them (Gruber 1993)—has been moving from the realm of Artificial-Intelligence laboratories to the desktops of domain experts. Ontologies have become common on the World-Wide Web. The ontologies on the Web range from large taxonomies categorizing Web sites (such as on Yahoo!) to categorizations of products for sale and their features (such as on Amazon.com). The WWW Consortium (W3C) is developing the Resource Description Framework (Brickley and Guha 1999), a language for encoding knowledge on Web pages to make it understandable to electronic agents searching for information. The Defense Advanced Research Projects Agency (DARPA), in conjunction with the W3C, is developing DARPA Agent Markup Language (DAML) by extending RDF with more expressive constructs aimed at facilitating agent interaction on the Web (Hendler and McGuinness 2000). Many disciplines now develop standardized ontologies that domain experts can use to share and annotate information in their fields. Medicine, for example, has produced large, standardized, structured vocabularies such as SNOMED (Price and Spackman 2000) and the semantic network of the Unified Medical Language System (Humphreys and Lindberg 1993). Broad general-purpose ontologies are emerging as well. For example, the United Nations Development Program and Dun & Bradstreet combined their efforts to develop the UNSPSC ontology which provides terminology for products and services (www.unspsc.org).

An ontology defines a common vocabulary for researchers who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them.

Why would someone want to develop an ontology? Some of the reasons are:

- To share common understanding of the structure of information among people or software agents
- To enable reuse of domain knowledge
- To make domain assumptions explicit
- To separate domain knowledge from the operational knowledge
- To analyze domain knowledge

Sharing common understanding of the structure of information among people or software agents is one of the more common goals in developing ontologies (Musen 1992; Gruber 1993). For example, suppose several different Web sites contain medical information or provide medical e-commerce services. If these Web sites share and publish the same underlying ontology of the terms they all use, then computer agents can extract and aggregate information from these different sites. The agents can use this aggregated information to answer user queries or as input data to other applications.

Enabling reuse of domain knowledge was one of the driving forces behind recent surge in ontology research. For example, models for many different domains need to represent the notion of time. This representation includes the notions of time intervals, points in time, relative measures of time, and so on. If one group of researchers develops such an ontology in detail, others can simply reuse it for their domains. Additionally, if we need to build a large

Protégé

https://protege.stanford.edu/support.php

DOCUMENTATION

Learn about the Protégé toolset and developing ontologies using our product documentation.

USERS | DEVELOPERS

Look here first!

» *Protégé Frequently Asked Questions*
Find answers to commonly asked questions about the Protégé Project.

» *WebProtégé Overview*
Get an overview of WebProtégé features and the latest project activity.

» *WebProtégé User's Guide*
A short user's guide with screencasts and screenshots describing the main user interface elements in WebProtégé.

» *Protégé Desktop*
Top-level entry point for all documentation pertaining to the latest version of Protégé Desktop, including an overview of editor features, links to tutorials etc.

» *Ontology Development 101*
A guide to creating your first ontology.

» *Protégé OWL Tutorial*
A substantial guide to the Web Ontology Language (OWL) and ontology engineering.

A Practical Guide To Building OWL Ontologies
Using Protégé 4 and CO-ODE Tools
Edition 1.3

Matthew Horridge

Contributors

v 1.0 Holger Knublauch , Alan Rector , Robert Stevens , Chris Wroe
v 1.1 Simon Jupp, Georgina Moulton, Robert Stevens
v 1.2 Nick Drummond, Simon Jupp, Georgina Moulton, Robert Stevens
v 1.3 Sebastian Brandt

The University Of Manchester

March 24, 2011

Protégé

http://protegeproject.github.io/protege/

DOCUMENTATION

Learn about the Protégé toolset and developing ontologies using our product documentation.

USERS | DEVELOPERS

Look here first!

» *Protégé Frequently Asked Questions*
Find answers to commonly asked questions about the Protégé Project.

» *WebProtégé Overview*
Get an overview of WebProtégé features and the latest project activity.

» *WebProtégé User's Guide*
A short user's guide with screencasts and screenshots describing the main user interface elements in WebProtégé.

» *Protégé Desktop*
Top-level entry point for all documentation pertaining to the latest version of Protégé Desktop, including an overview of editor features, links to tutorials etc.

» *Ontology Development 101*
A guide to creating your first ontology.

» *Protégé OWL Tutorial*
A substantial guide to the Web Ontology Language (OWL) and ontology engineering.

Protégé 5 Documentation

Installation | Getting Started | Views | Menus | Class Expression Syntax

Protege Documentation

This is the official documentation for Protégé 5.5.0. You can find information about the Protégé user interface including descriptions of the various views and menu items.

Installation
Explains how to install Protégé on Windows, Mac OS X and Linux.

Getting Started
A quick start guide for Protege.

Views
Provides a list of all of the default views that are distributed with Protégé

Menus
Explains what each menu item in Protégé does

Class Expression Syntax
Provides a reference for the class expression syntax that is used throughout Protégé.

https://protegewiki.stanford.edu/wiki/Pr4_UG_mi_Glossary#Ontology

Ontology

An ontology is a formally-defined vocabulary for a particular domain of interest. Ontologies are typically based on a class hierarchy (asserted and/or inferred), supplemented by assorted properties.

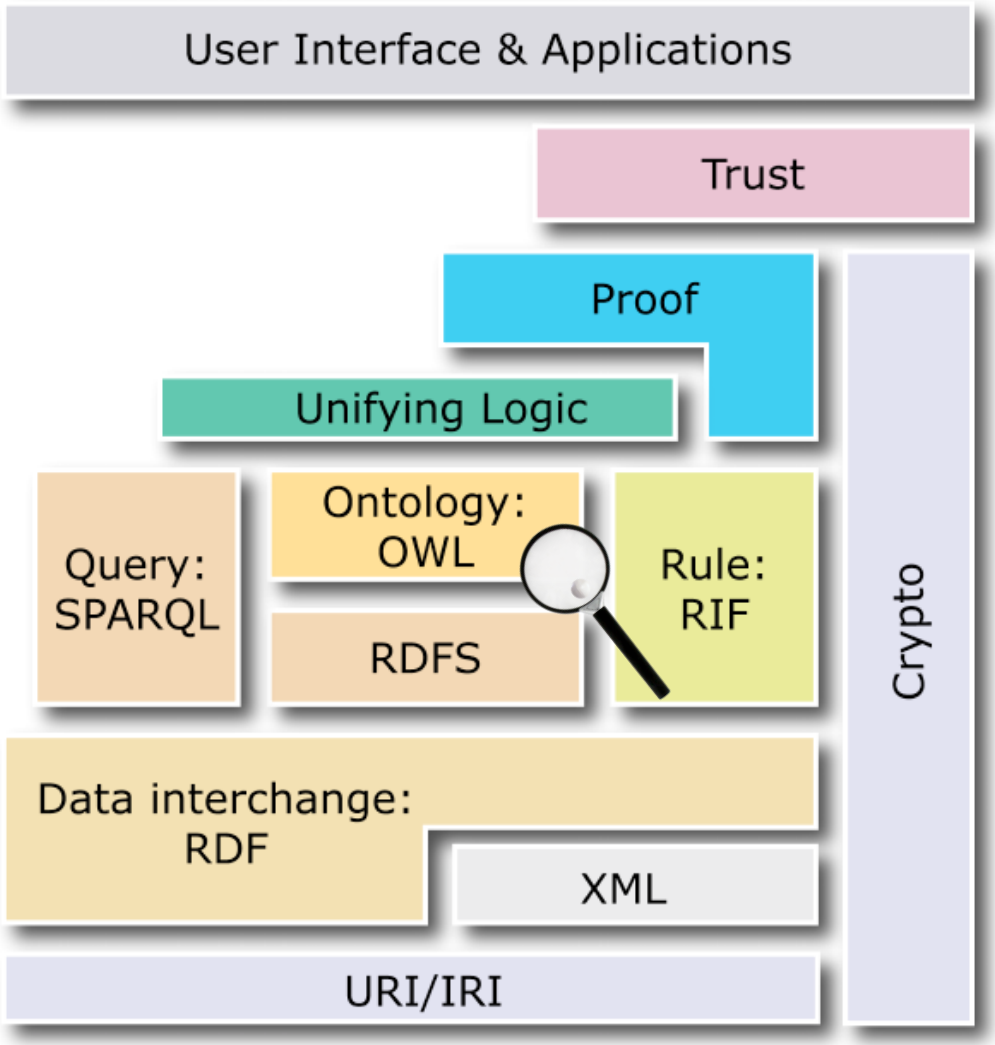
Open World Assumption

The Open World Assumption, used by OWL, says that "just because we don't know something to be true does not mean that we can assume it to be false"

OWL

OWL provides the theoretical basis for Protege 4 ontologies

OWL & the Semantic Web Architecture



What are OWL Ontologies?

Ontologies are used to capture knowledge about some domain of interest. An ontology describes the concepts in the domain and also the relationships that hold between those concepts.

Different ontology languages provide different facilities. The most recent development in standard ontology languages is OWL from the World Wide Web Consortium (W3C)

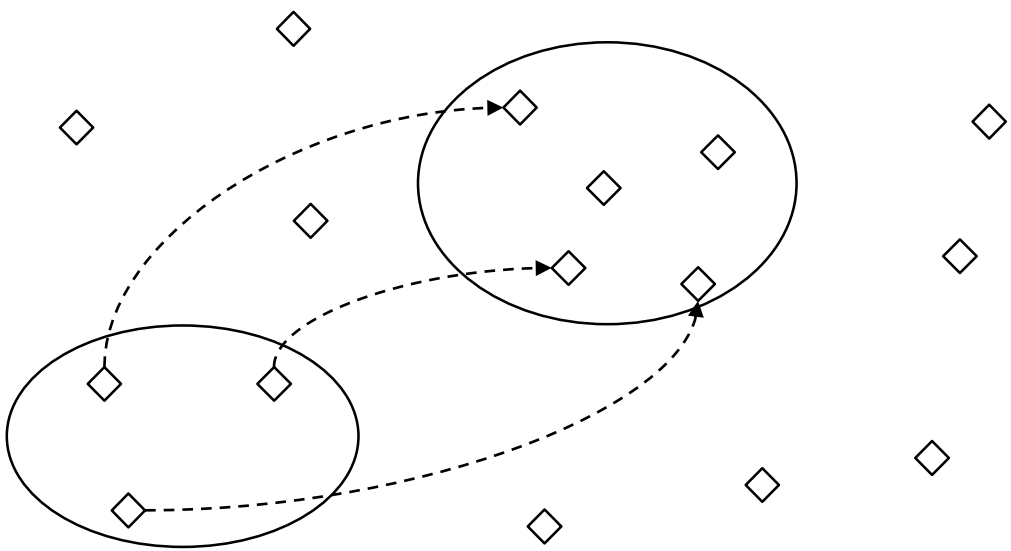
The logical model allows the use of a reasoner which can check whether or not all of the statements and definitions in the ontology are mutually consistent and can also recognise which concepts fit under which definitions. The reasoner can therefore help to maintain the hierarchy correctly.

A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools - Edition 1.3

Main ideas of OWL (DL)?

Extensional Logic

1. Organising the objects with populate the world into classes according to the relationships that linked objects together



2. An object is not defined by its “nature“, but through its relationships with other objects

Components of OWL Ontologies: Individuals

1) Individuals

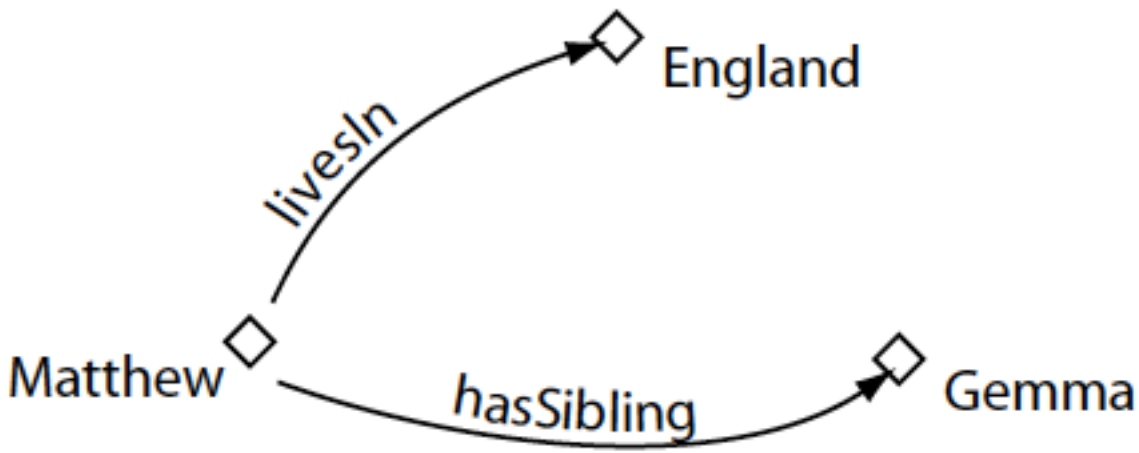
Individuals, represent objects in the domain in which we are interested

<u>Terminology</u>: « individual », « instance », « object »

Components of OWL Ontologies: Properties

2) Properties

Properties are binary relations on individuals, i.e. properties link two individuals together.

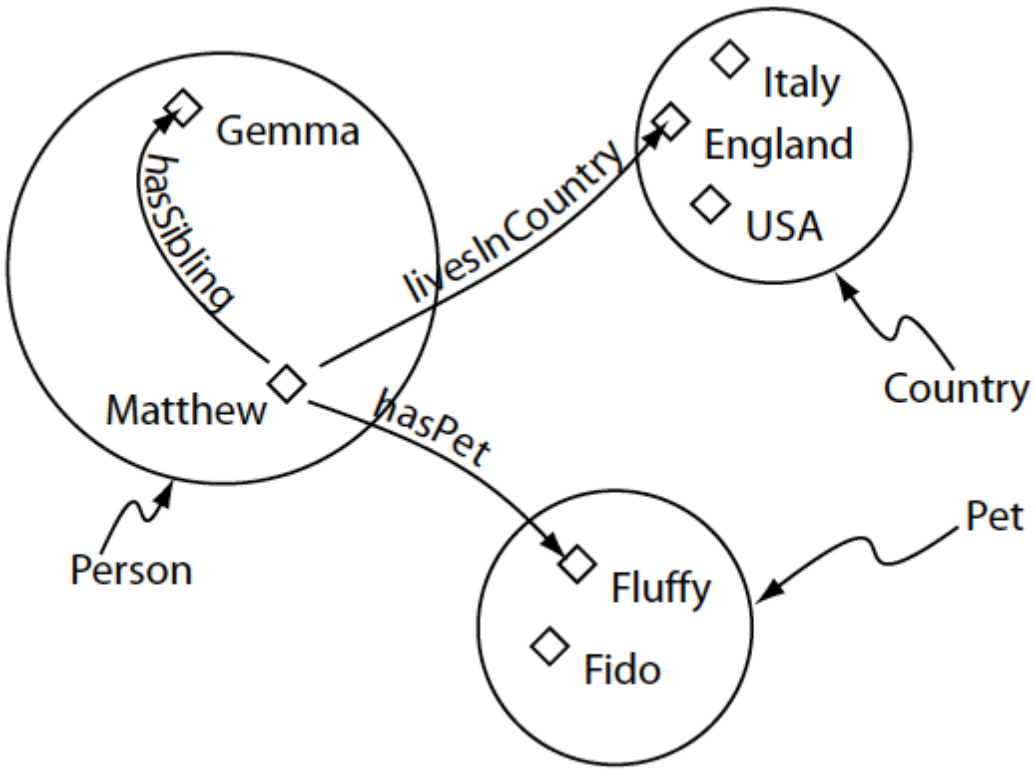


<u>Terminology</u>: « properties », « slots » (Protégé), « roles » (DL), « relations », « attributes »

Components of OWL Ontologies: Classes

3) Classes

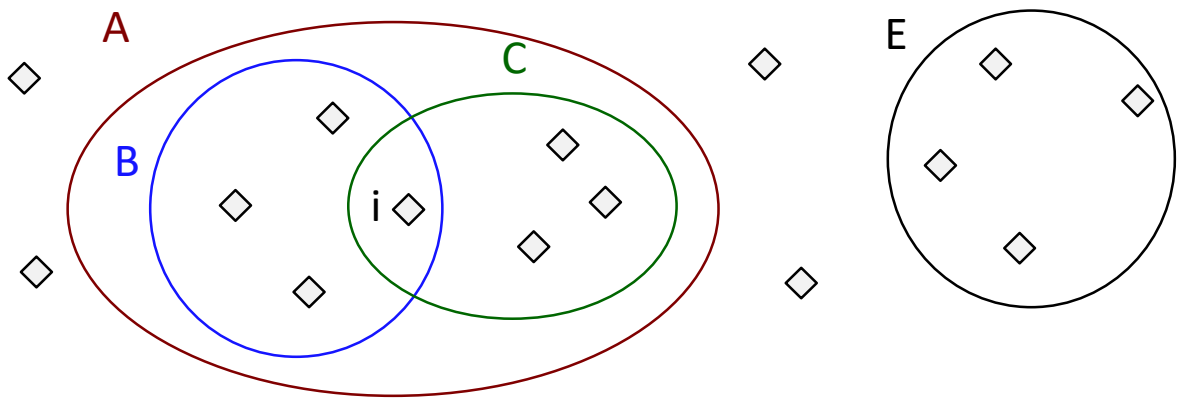
OWL classes are interpreted as sets that contain individuals.



Classes are a concrete representation of concepts.

Components of OWL Ontologies: Classes

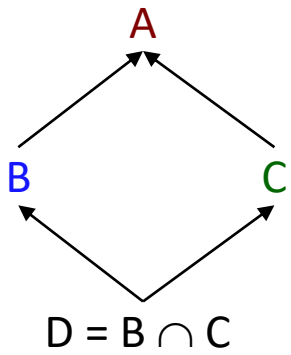
An individual can belong to different classes: $i \in B$, $i \in C$, $i \in B \cap C$



As sets, classes can be defined using set operators : $\cup$, $\cap$ $A = B \cup C$

Classes may be organised into a superclass-subclass hierarchy corresponding to inclusion between sets:

$B \subseteq A$ All members of the class B are members of the class A



Classes can be disjoint

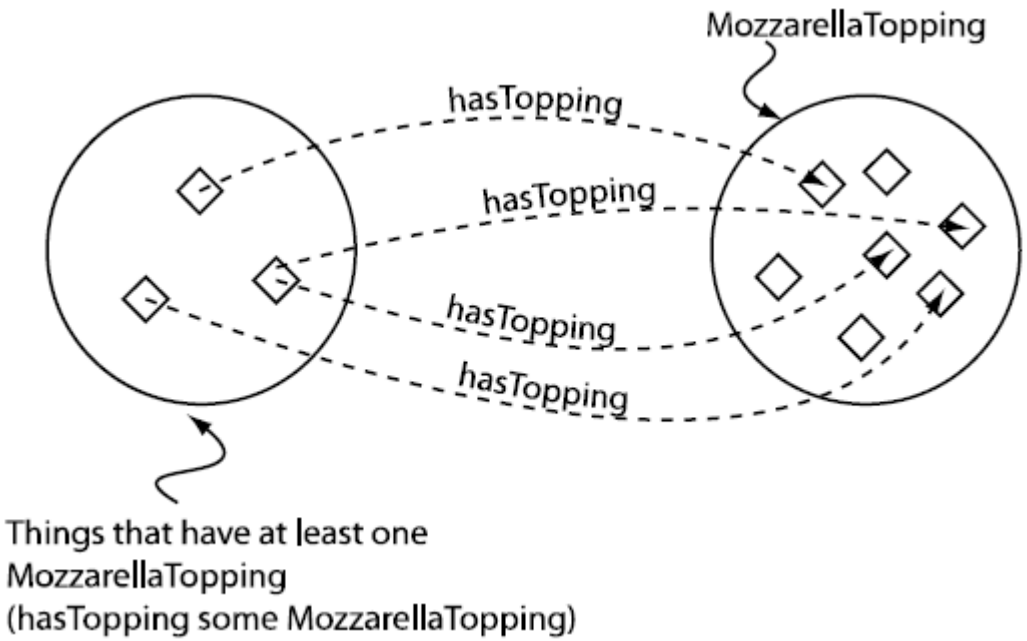
Components of OWL Ontologies: Properties Restriction

Classes are defined (described?) using formal descriptions that state precisely the requirements for membership of the class.

4) Property restriction A means to define classes of individuals

a) **Existential Restrictions**: describes (anonymous) classes of individuals that participate in *at least one* (some) relationship along a specified property to individuals that are members of a specified class.

(at least one value of the property must be of a certain type)



Components of OWL Ontologies: Properties restriction

b) Universal Restrictions: describes (anonymous) classes of individuals that for a given property *only* (only) have relationships along this property to individuals that are members of a specified class.

(all values of the property must be of a certain type)

c) Has value: at least one of the values of the property is a certain value

Components of OWL Ontologies: Reasoner

5) Reasoner

an inconsistent class is a class which cannot contain any individual because of its definition

<u>Terminology</u>: « classes », « concepts »

My 1st K-Graph in Protégé

椅子

armchair

chaise

fauteuil

pouf

banc

stool

长凳

tabouret

bench

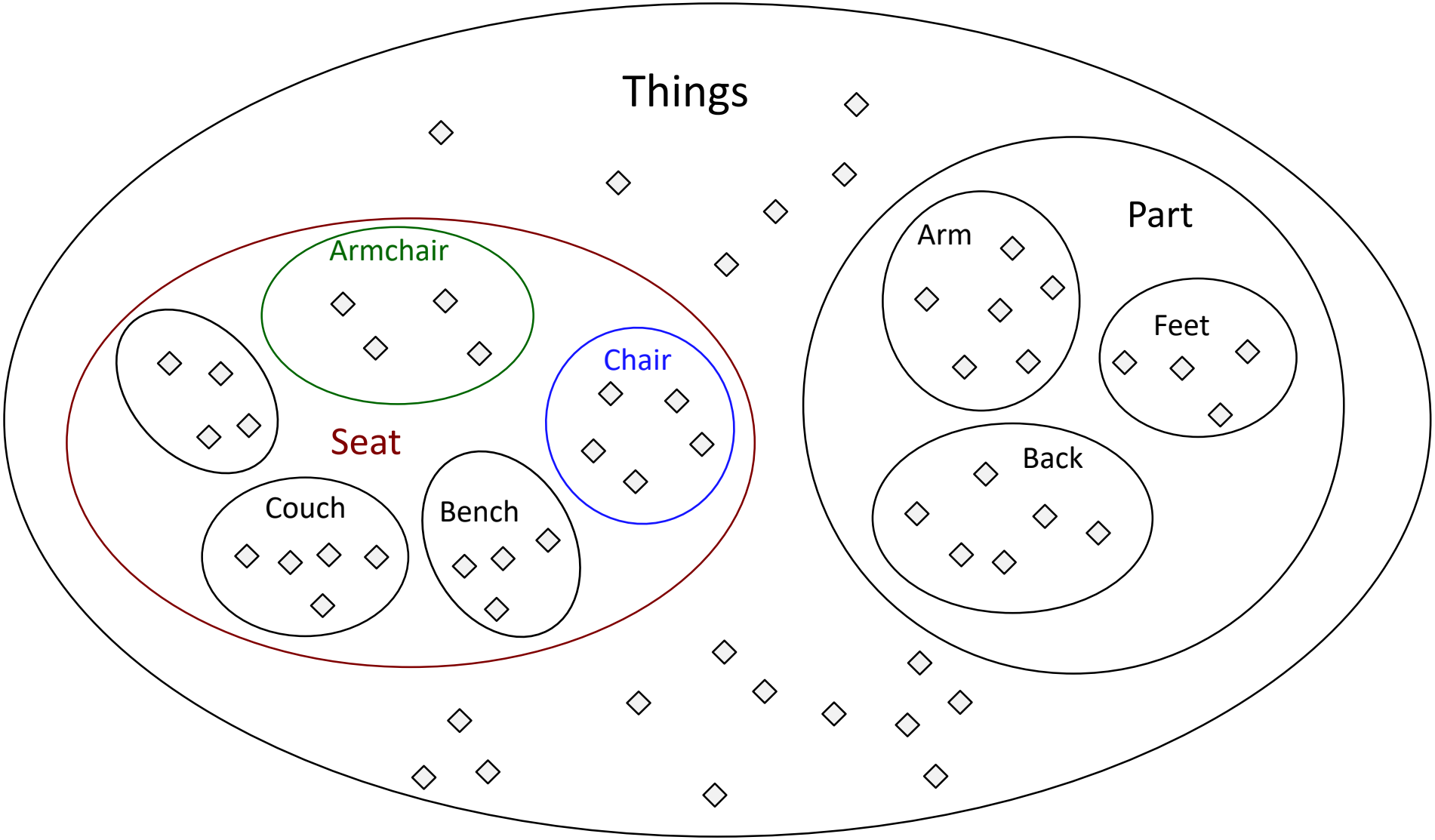
chair

canapé

couch

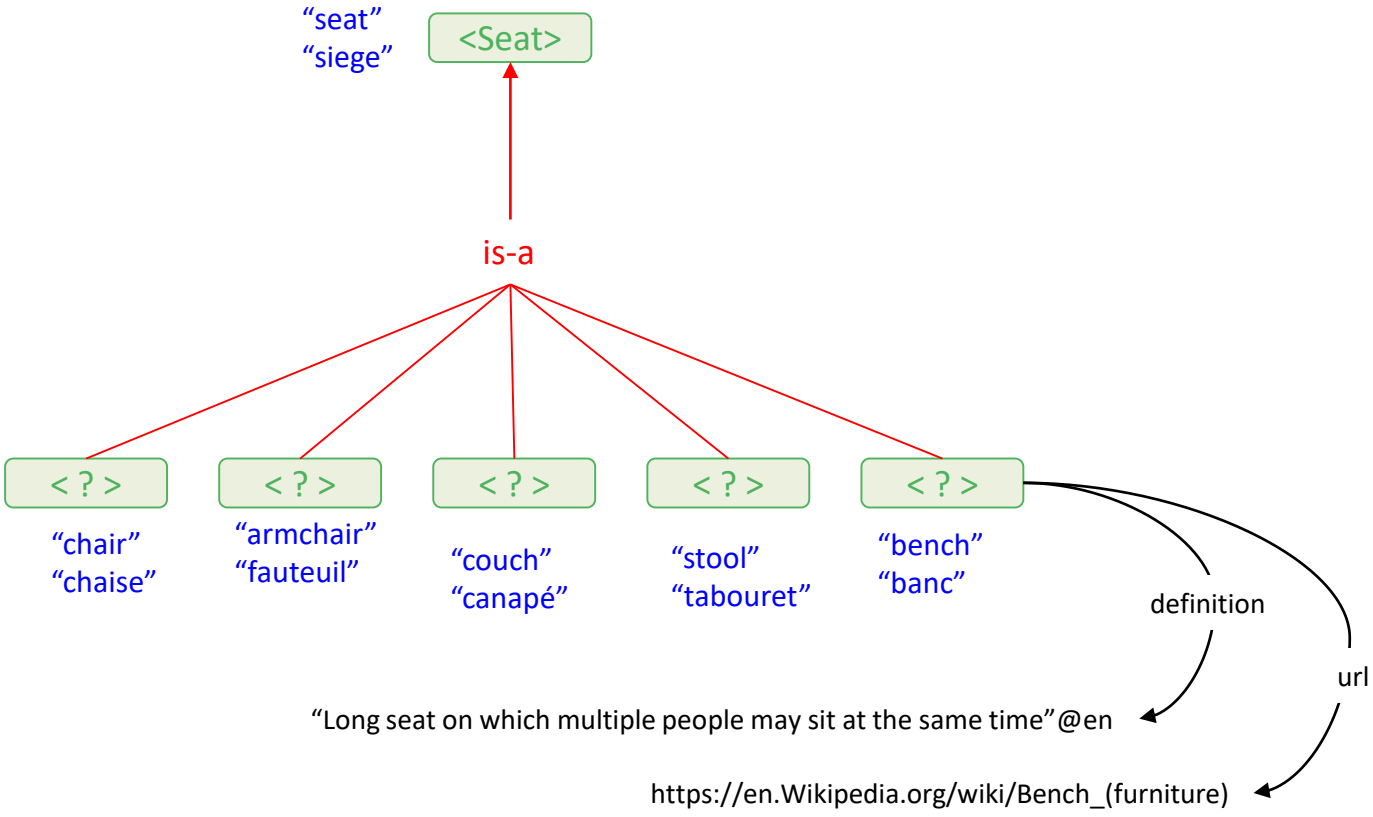
长椅

Building an OWL Ontology using Protégé: Named Classes



My 1st K-Graph in Protégé

1. Building my 1st K-Graph in Protégé
2. Querying my 1st K-Graph in Protégé

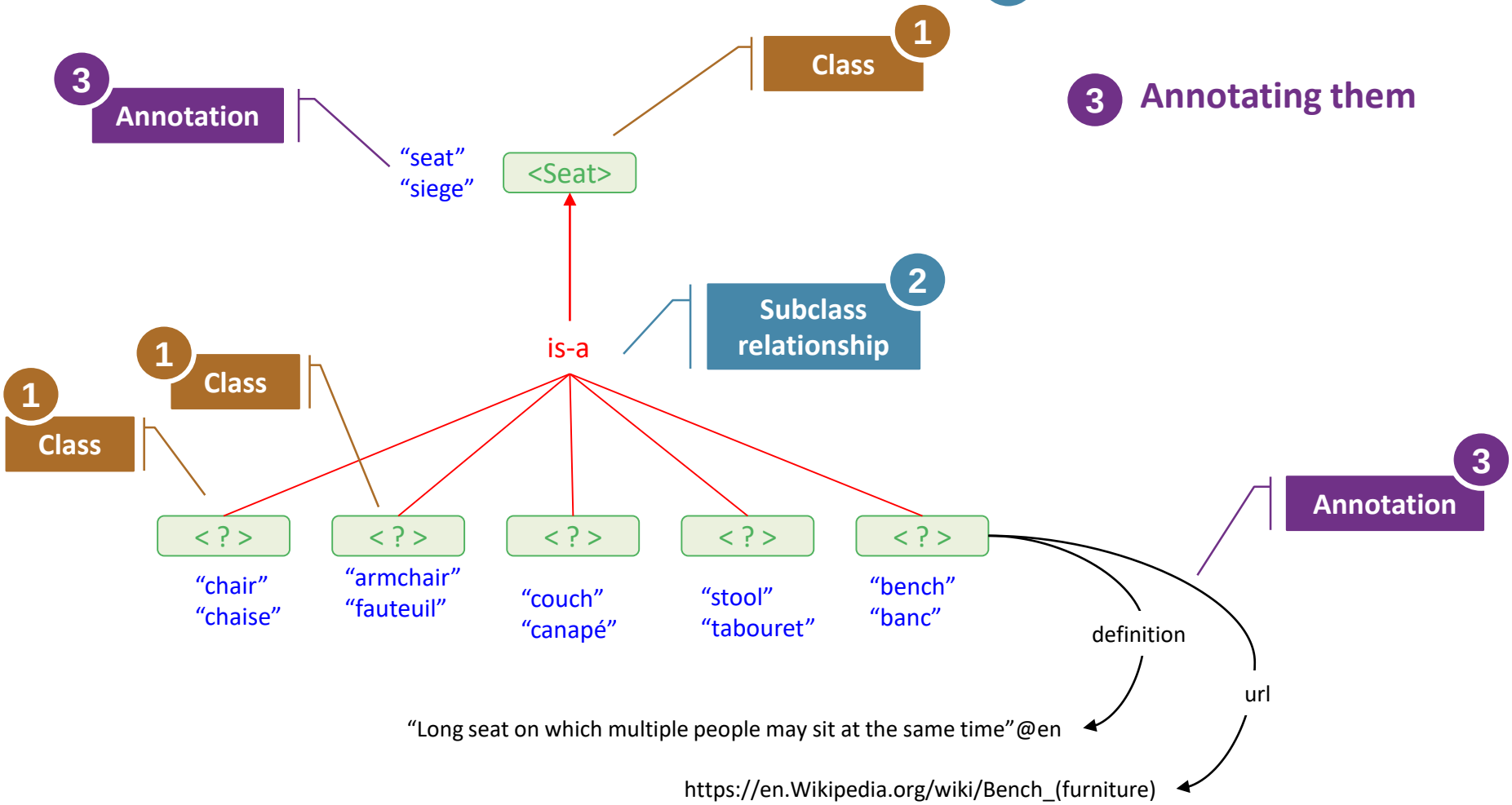


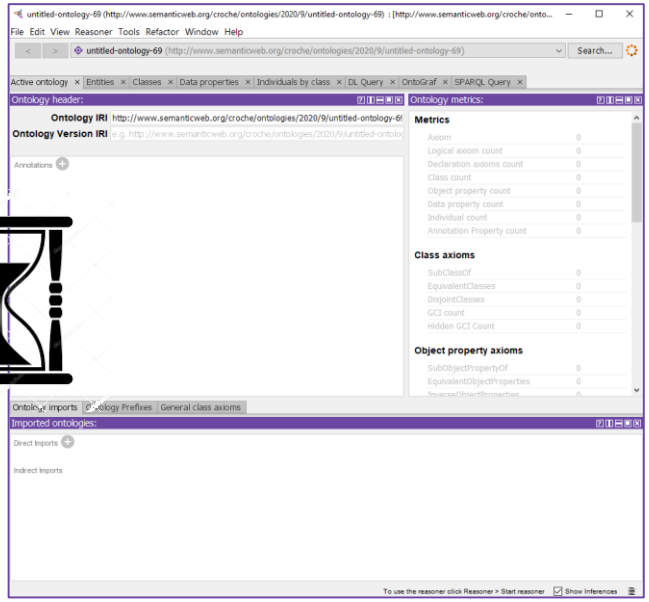
My 1st K-Graph in Protégé

1 Defining classes

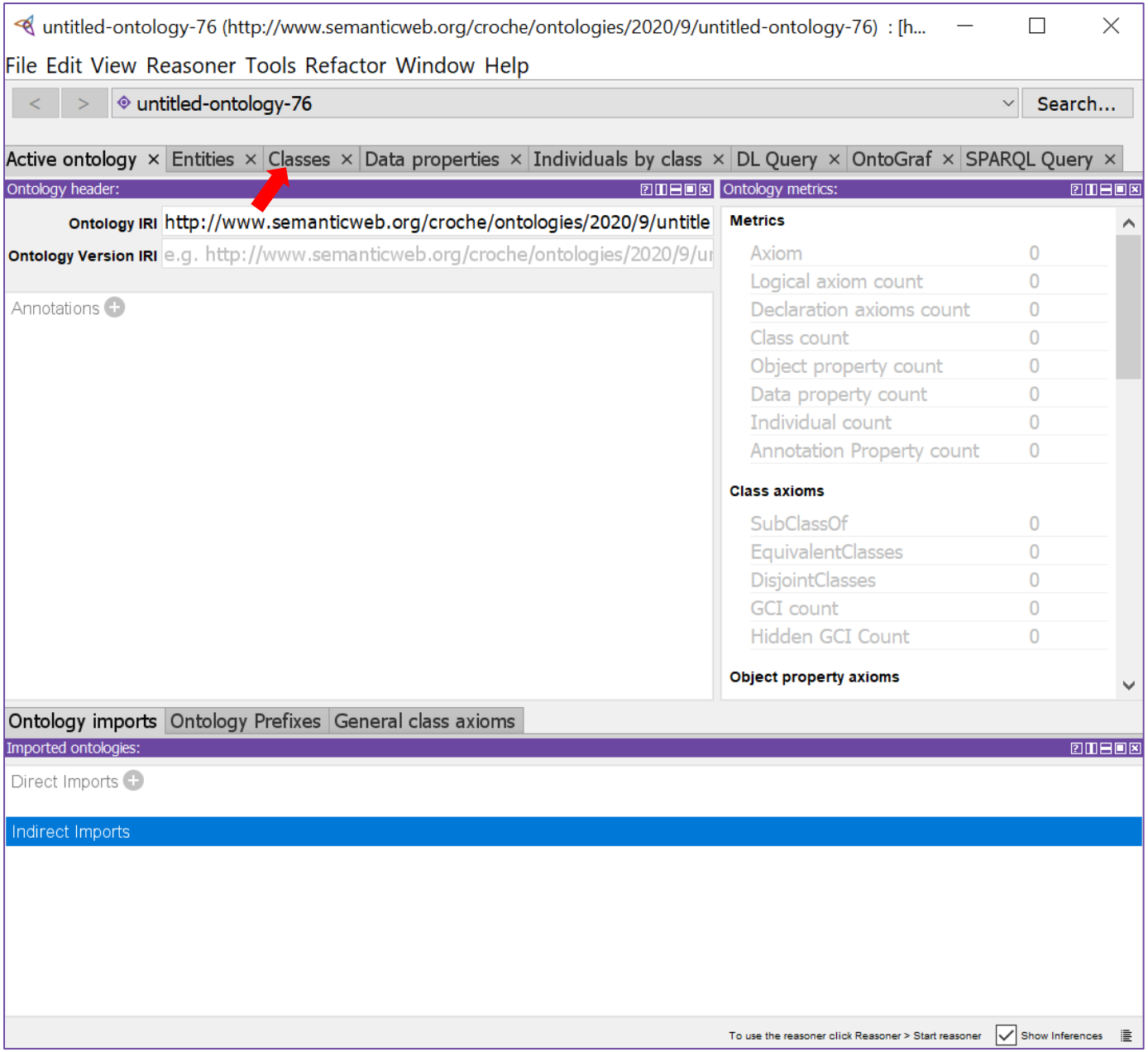
2 Organizing them

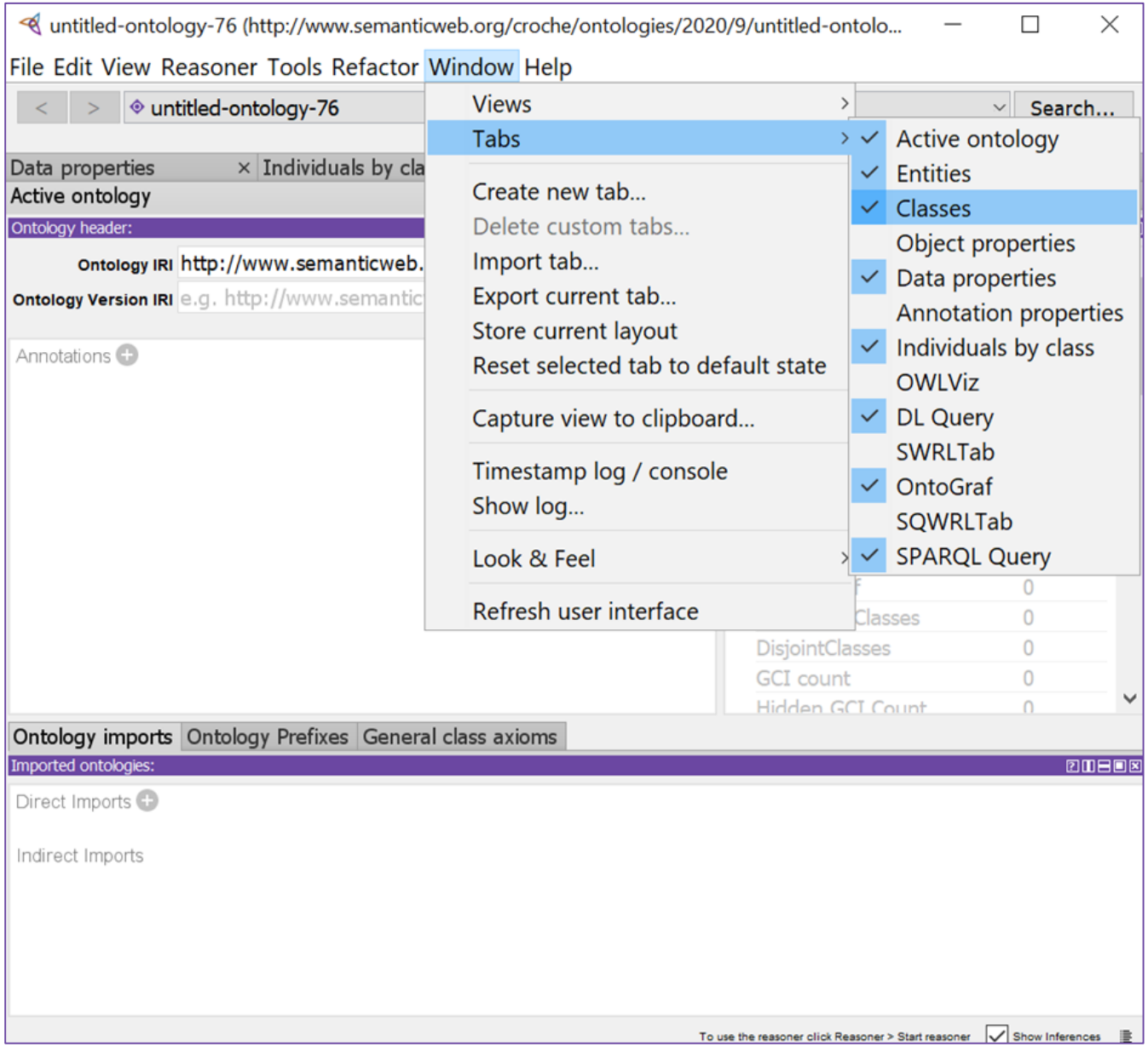
3 Annotating them




File Edit View Reasoner Tools Refactor Window Help
Search...
Active ontology
Entities
Classes
Data properties
Individuals by class
DL Query
OntoGraf
SPARQL Query
Ontology header:
Ontology IRI
Ontology Version IRI
Annotations
Ontology metrics:
Metrics
Class axioms
Object property axioms
General class axioms
Imported ontologies:
Direct Imports
Indirect Imports
To use the reasoner click Reasoner > Start reasoner
Show Inferences

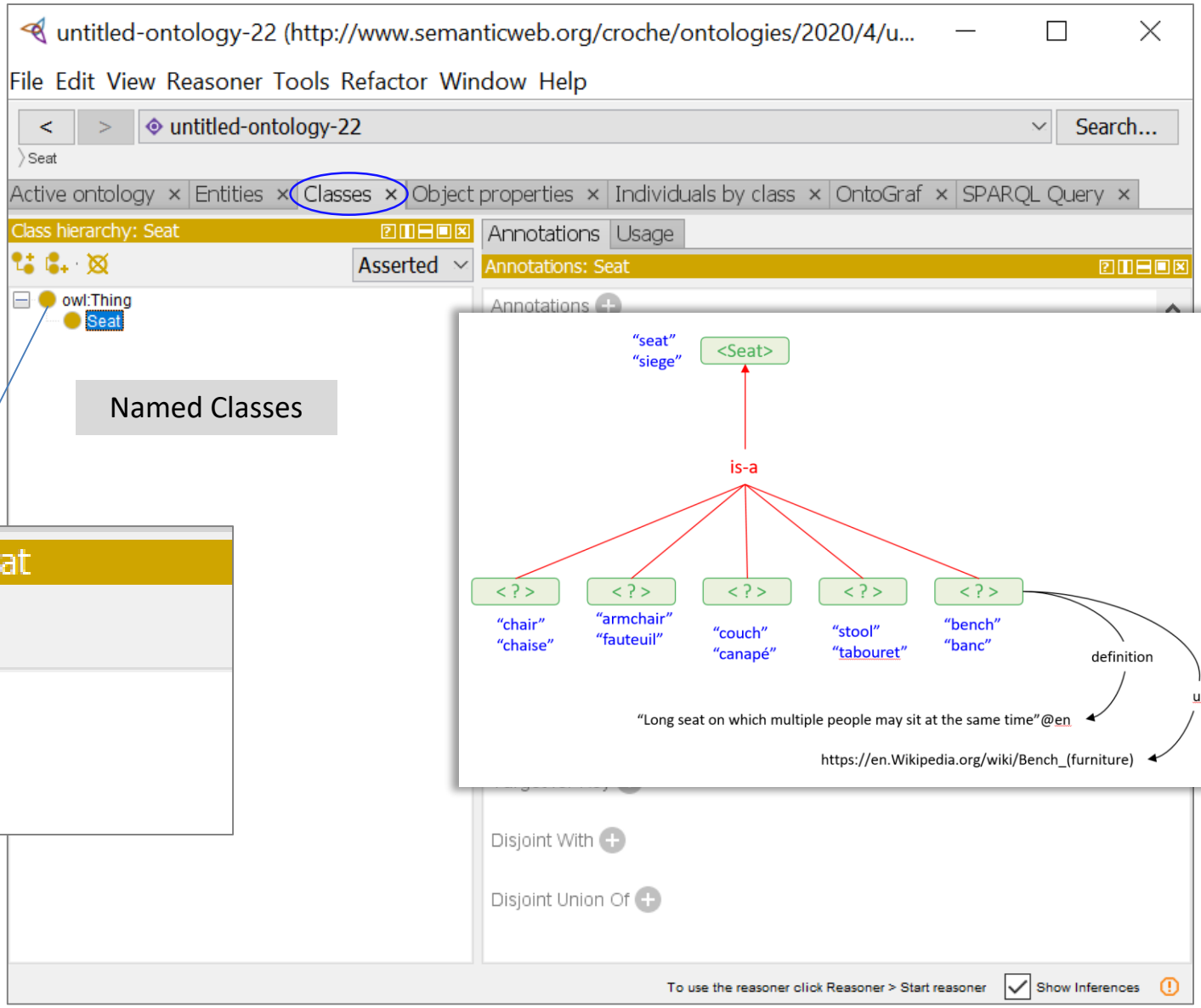
protégé


untitled-ontology-76 (http://www.semanticweb.org/croche/ontologies/2020/9/untitled-ontology-76) : [h...
File Edit View Reasoner Tools Refactor Window Help
untitled-ontology-76
Search...
Active ontology
Entities
Classes
Data properties
Individuals by class
DL Query
OntoGraf
SPARQL Query
Ontology header:
Ontology IRI
http://www.semanticweb.org/croche/ontologies/2020/9/untitle
Ontology Version IRI
e.g. http://www.semanticweb.org/croche/ontologies/2020/9/ur
Annotations
Ontology metrics:
Metrics
Axiom 0
Logical axiom count 0
Declaration axioms count 0
Class count 0
Object property count 0
Data property count 0
Individual count 0
Annotation Property count 0
Class axioms
SubClassOf 0
EquivalentClasses 0
DisjointClasses 0
GCI count 0
Hidden GCI Count 0
Object property axioms
Ontology imports
Ontology Prefixes
General class axioms
Imported ontologies:
Direct Imports
Indirect Imports
To use the reasoner click Reasoner > Start reasoner
Show Inferences

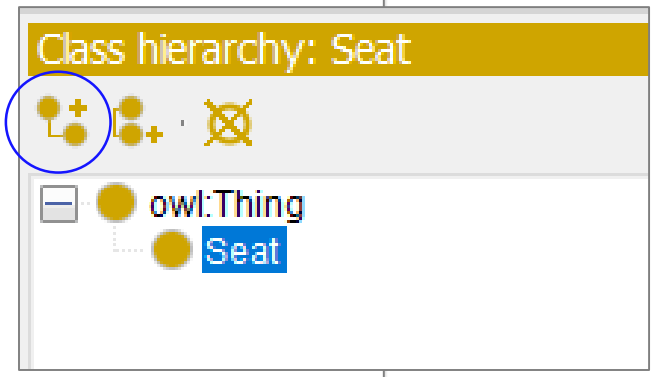

untitled-ontology-76 (http://www.semanticweb.org/croche/ontologies/2020/9/untitled-ontolo...
File Edit View Reasoner Tools Refactor Window Help
untitled-ontology-76
Search...
Data properties
Individuals by cla
Active ontology
Ontology header:
Ontology IRI http://www.semanticweb.
Ontology Version IRI e.g. http://www.semantic
Annotations
Views
Tabs
Create new tab...
Delete custom tabs...
Import tab...
Export current tab...
Store current layout
Reset selected tab to default state
Capture view to clipboard...
Timestamp log / console
Show log...
Look & Feel
Refresh user interface
Active ontology
Entities
Classes
Object properties
Data properties
Annotation properties
Individuals by class
OWLViz
DL Query
SWRLTab
OntoGraf
SQWRLTab
SPARQL Query
Classes 0
DisjointClasses 0
GCI count 0
Ontology imports
Ontology Prefixes
General class axioms
Imported ontologies:
Direct Imports
Indirect Imports
To use the reasoner click Reasoner > Start reasoner
Show Inferences

1 Defining classes 2 Organizing them

The class Thing represents the set containing all individuals

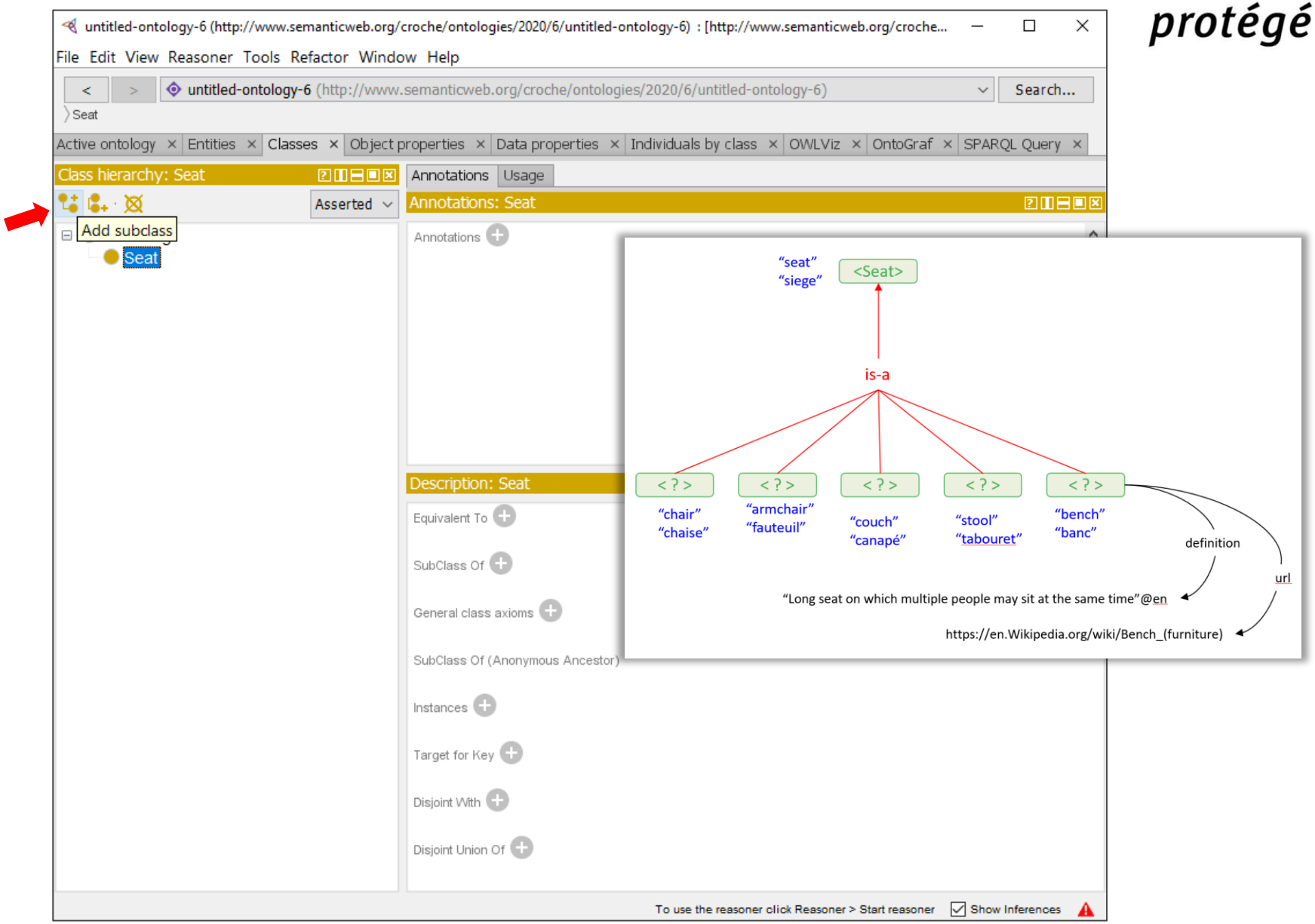


Named Classes



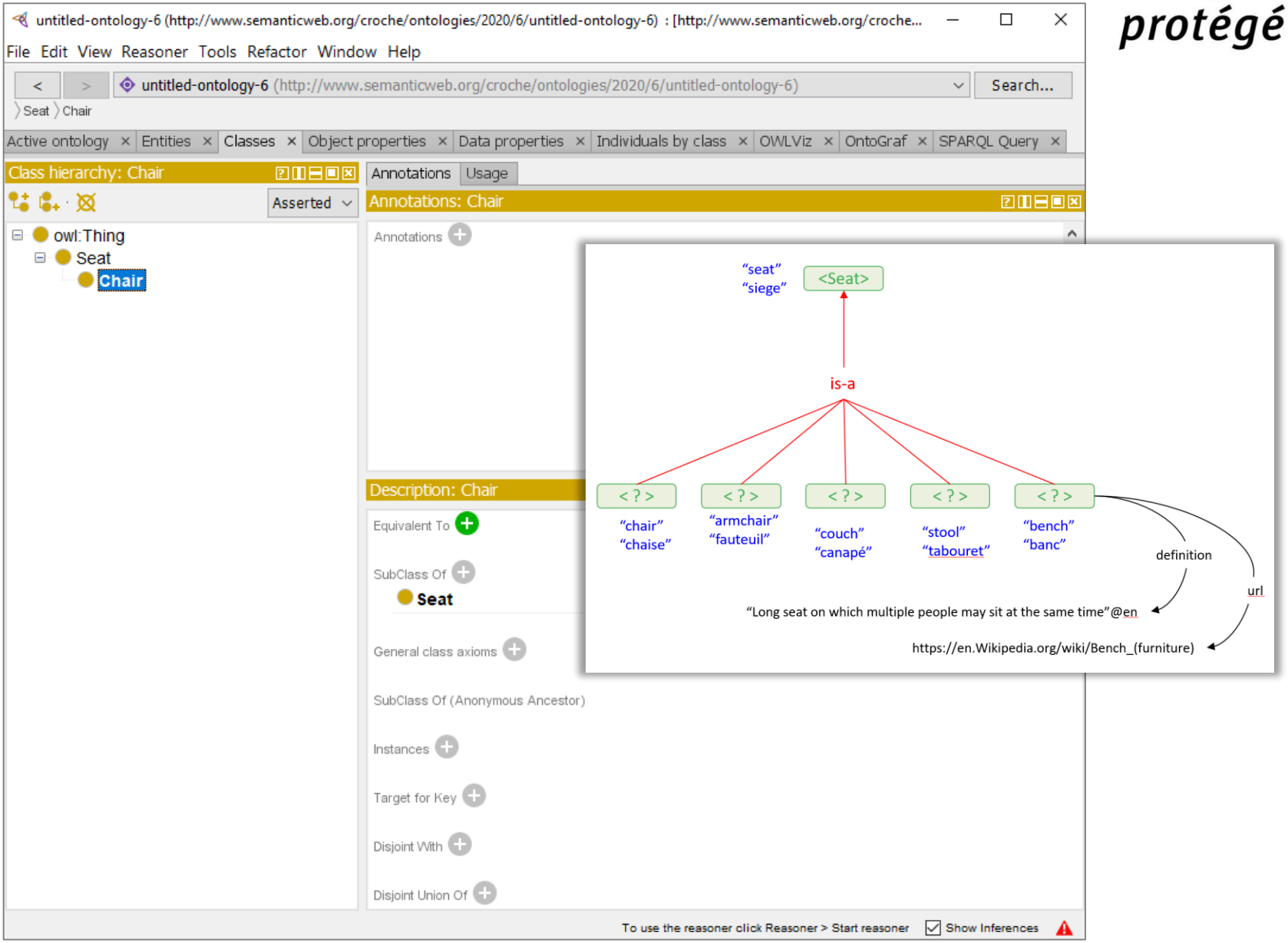
1 Defining classes

2 Organizing them



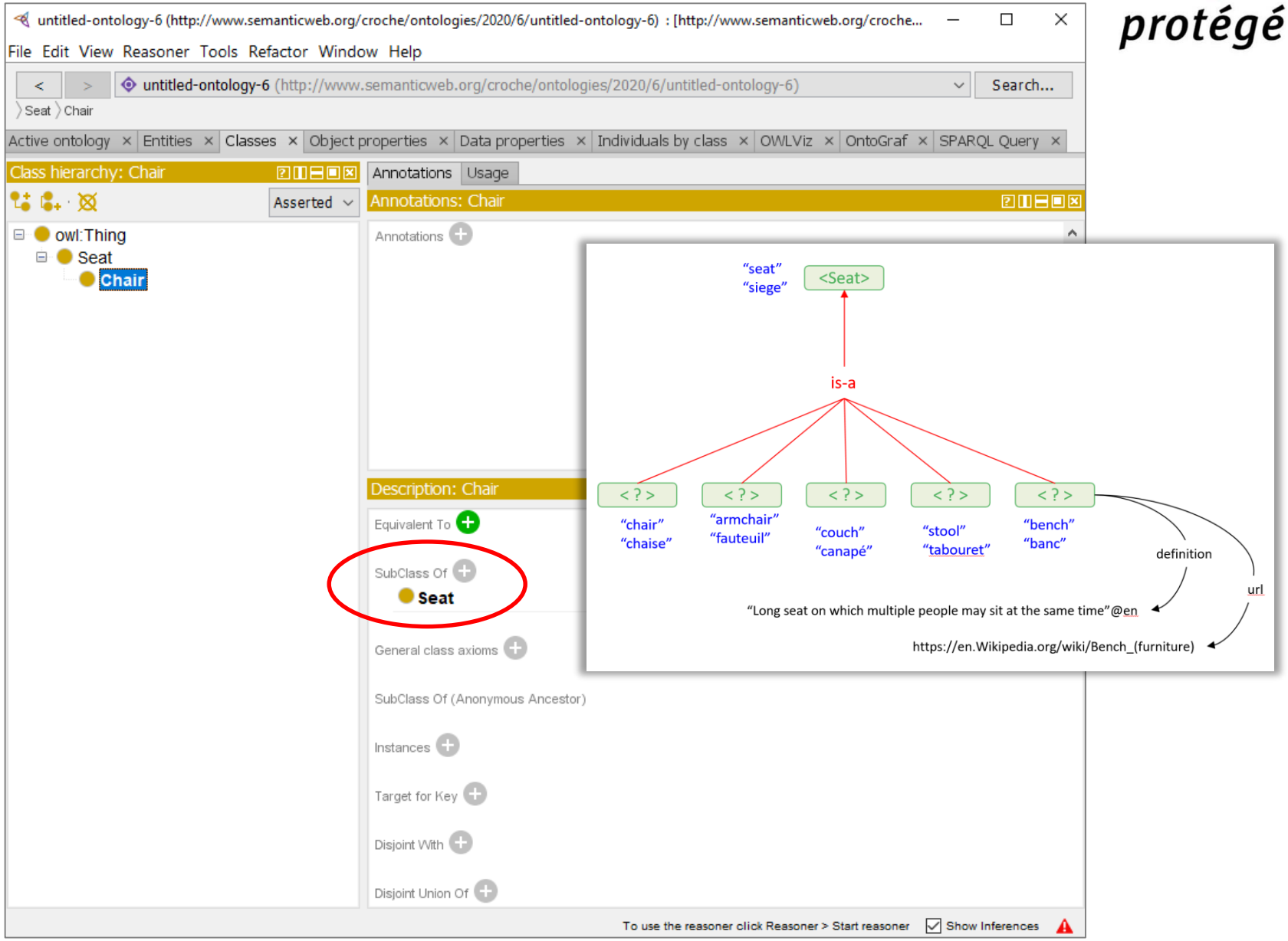
1 Defining classes

2 Organizing them



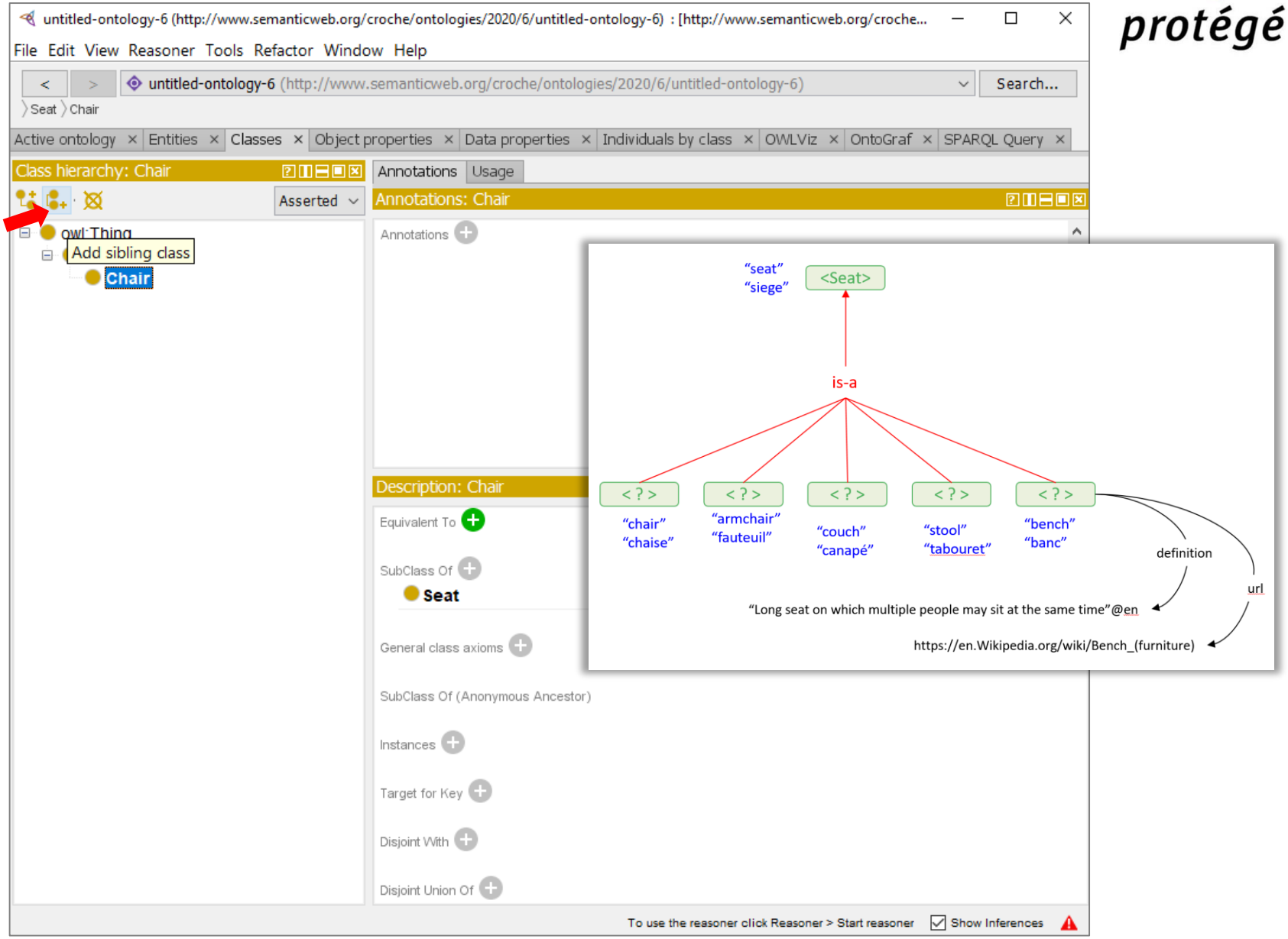
1 Defining classes

2 Organizing them



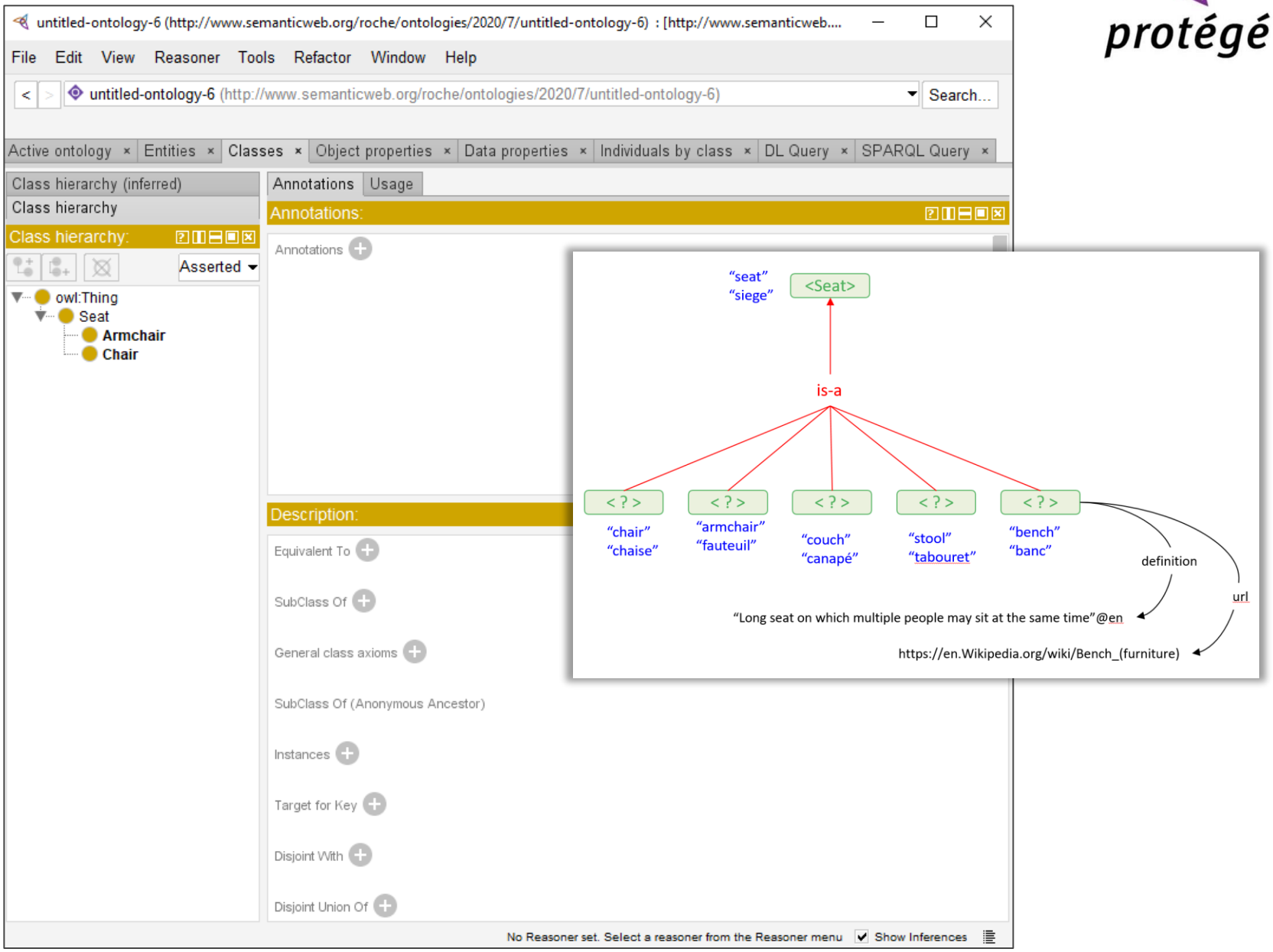
1 Defining classes

2 Organizing them



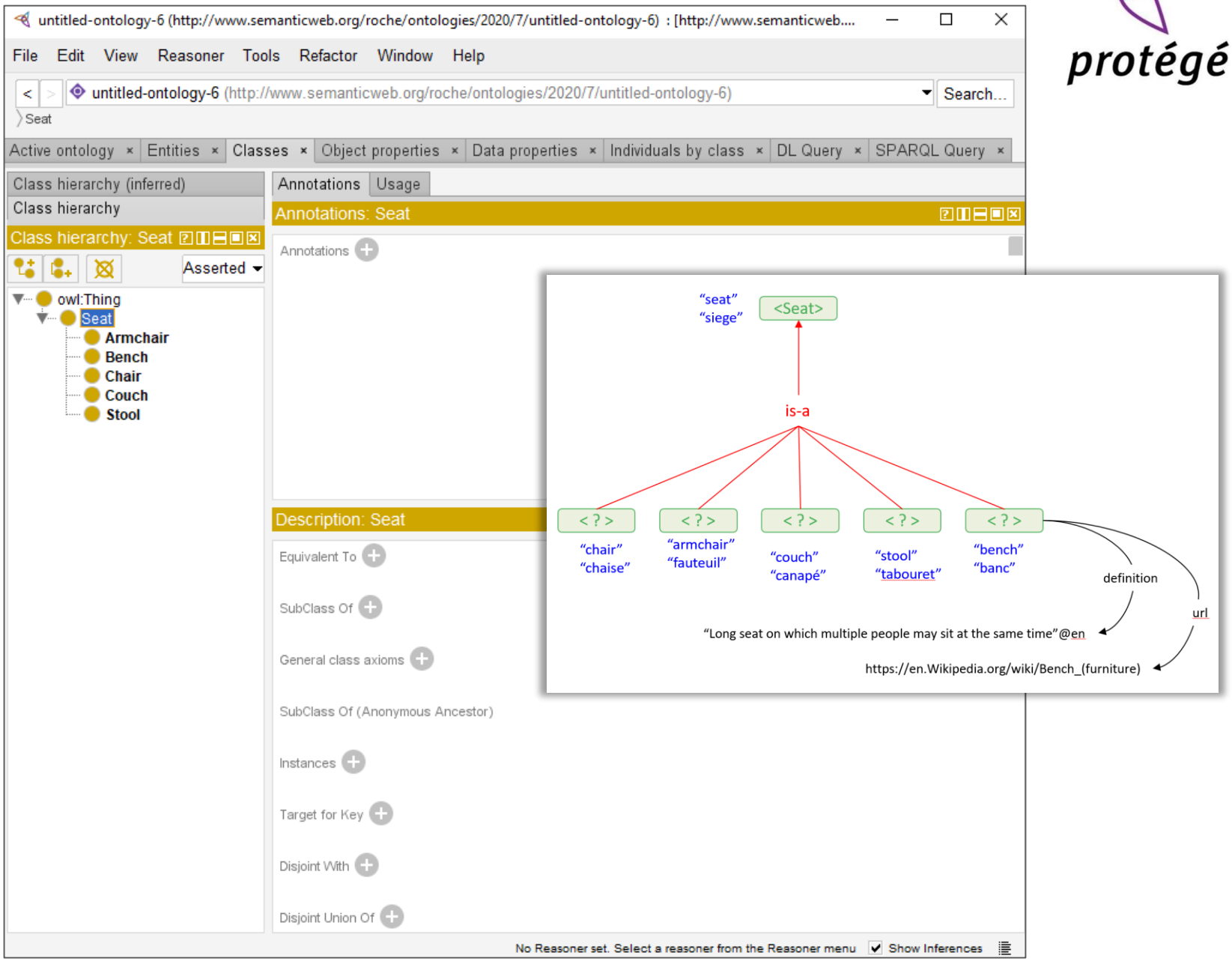
1 Defining classes

2 Organizing them

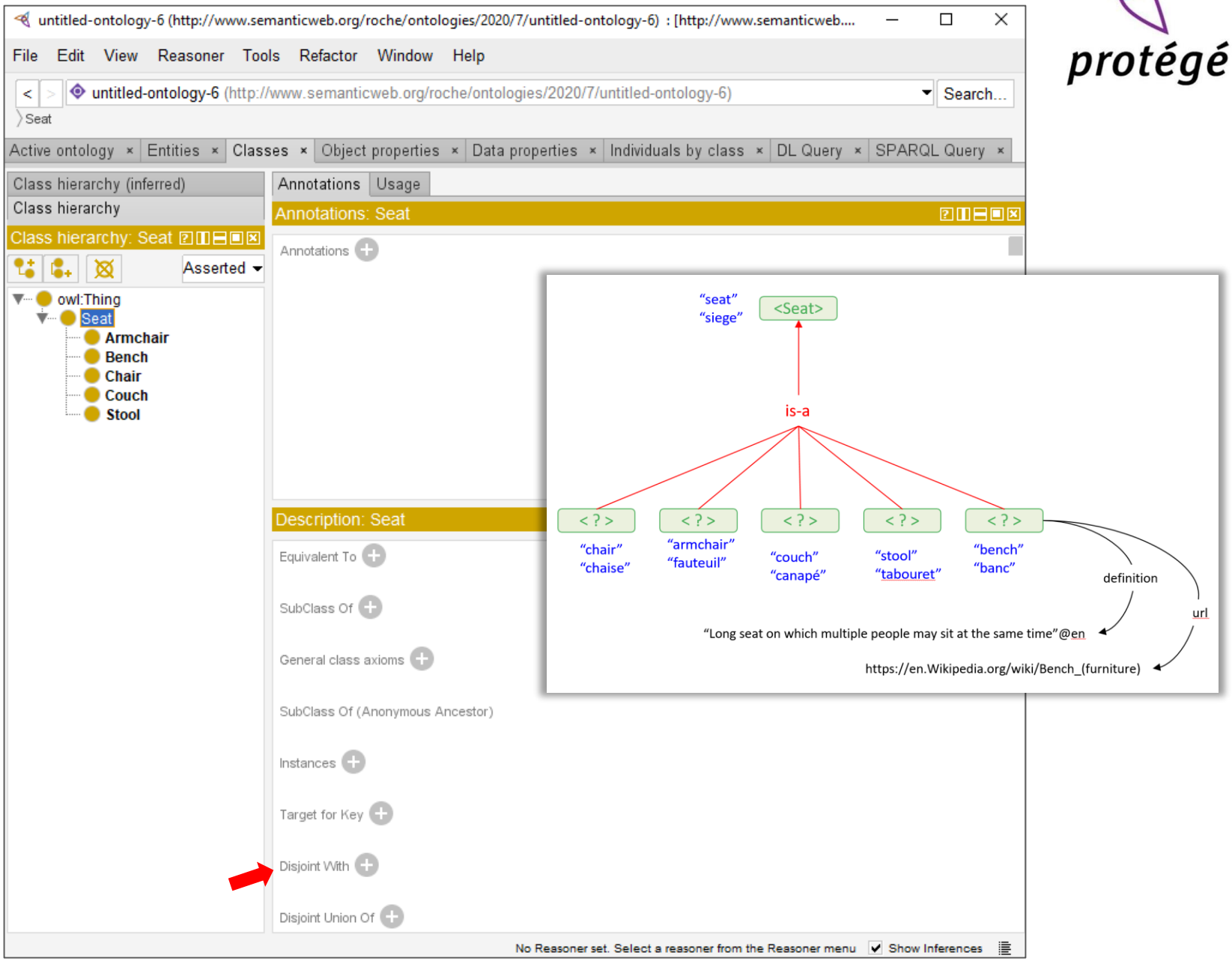


1 Defining classes

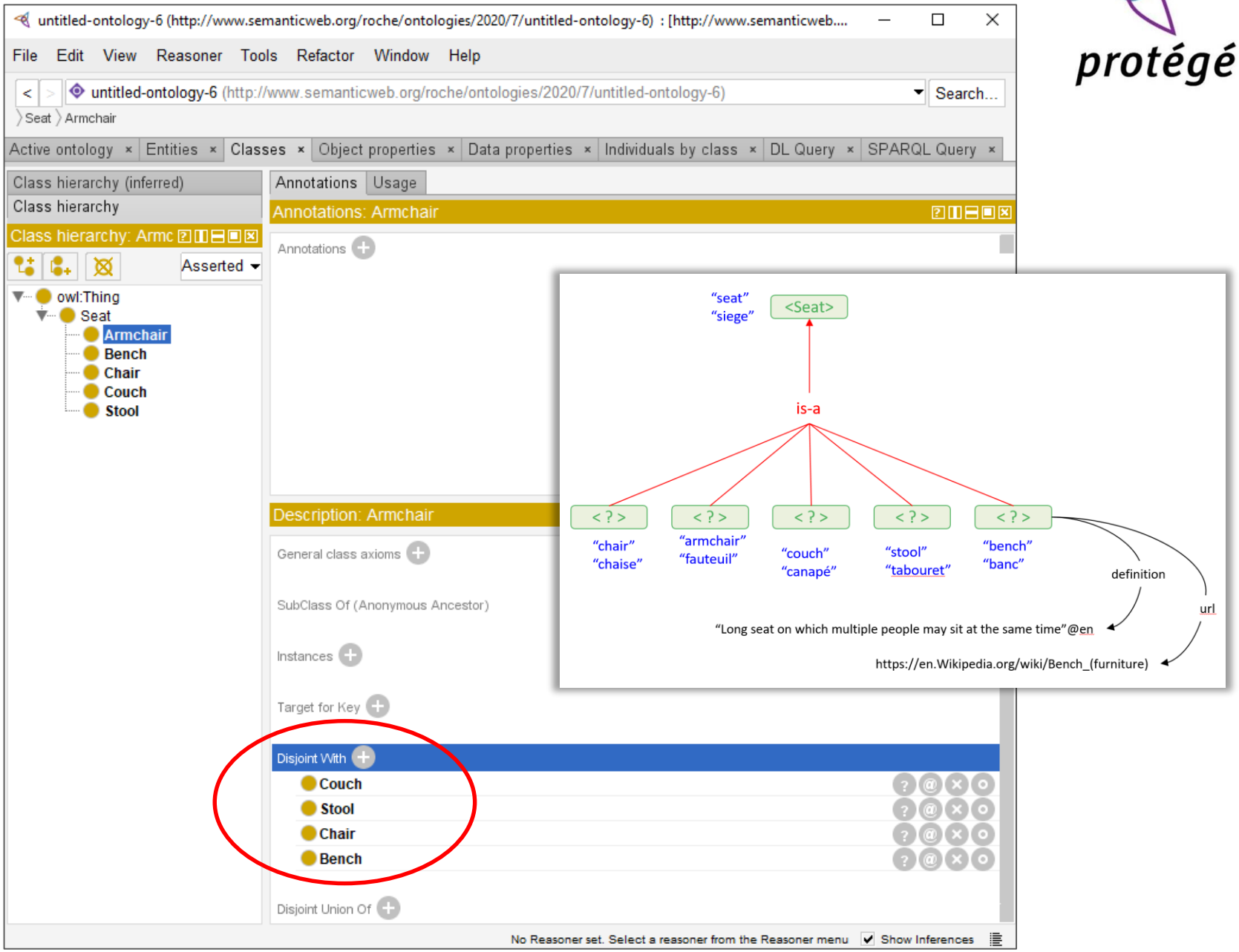
2 Organizing them



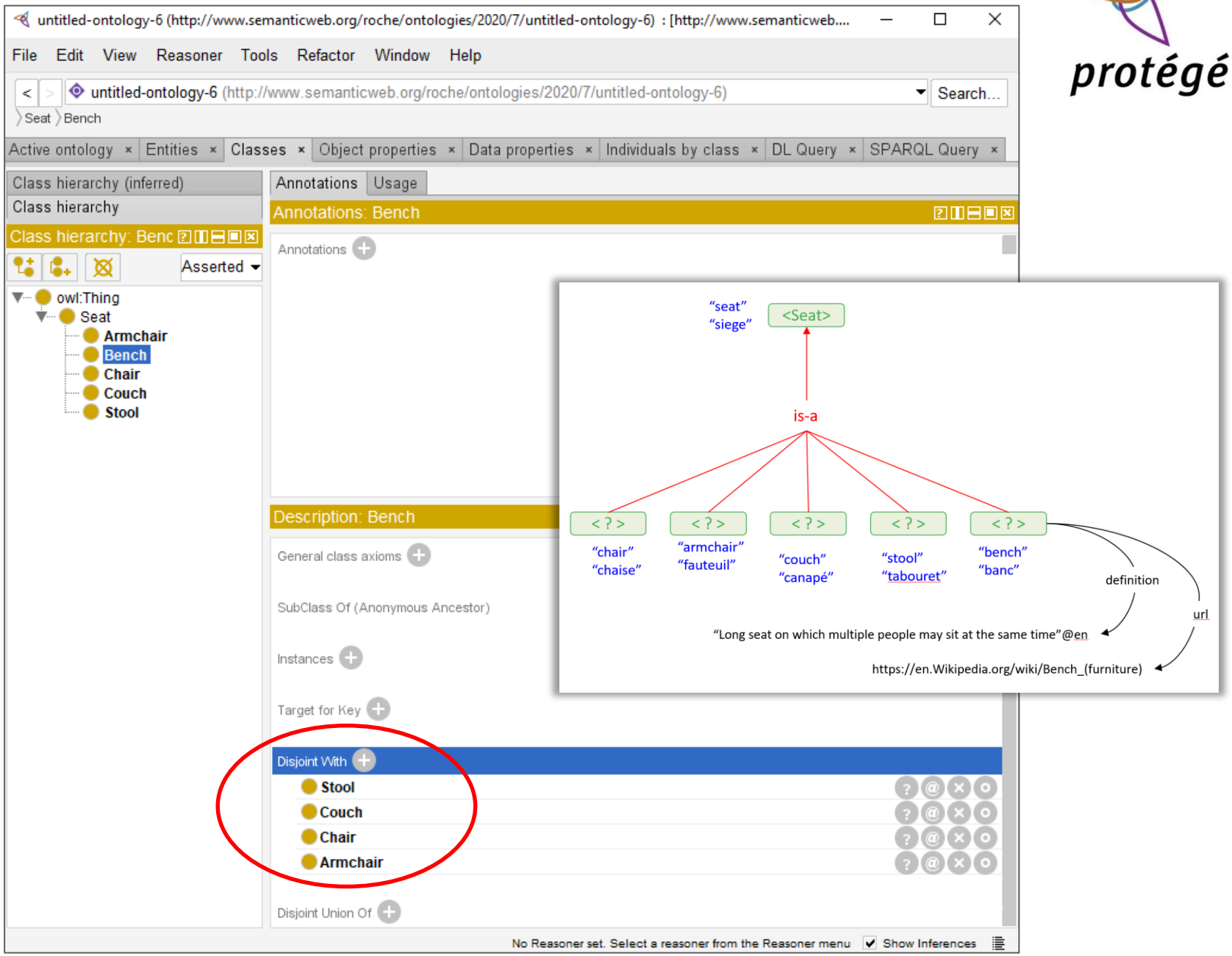
1 Defining classes 2 Organizing them 2' Adding properties



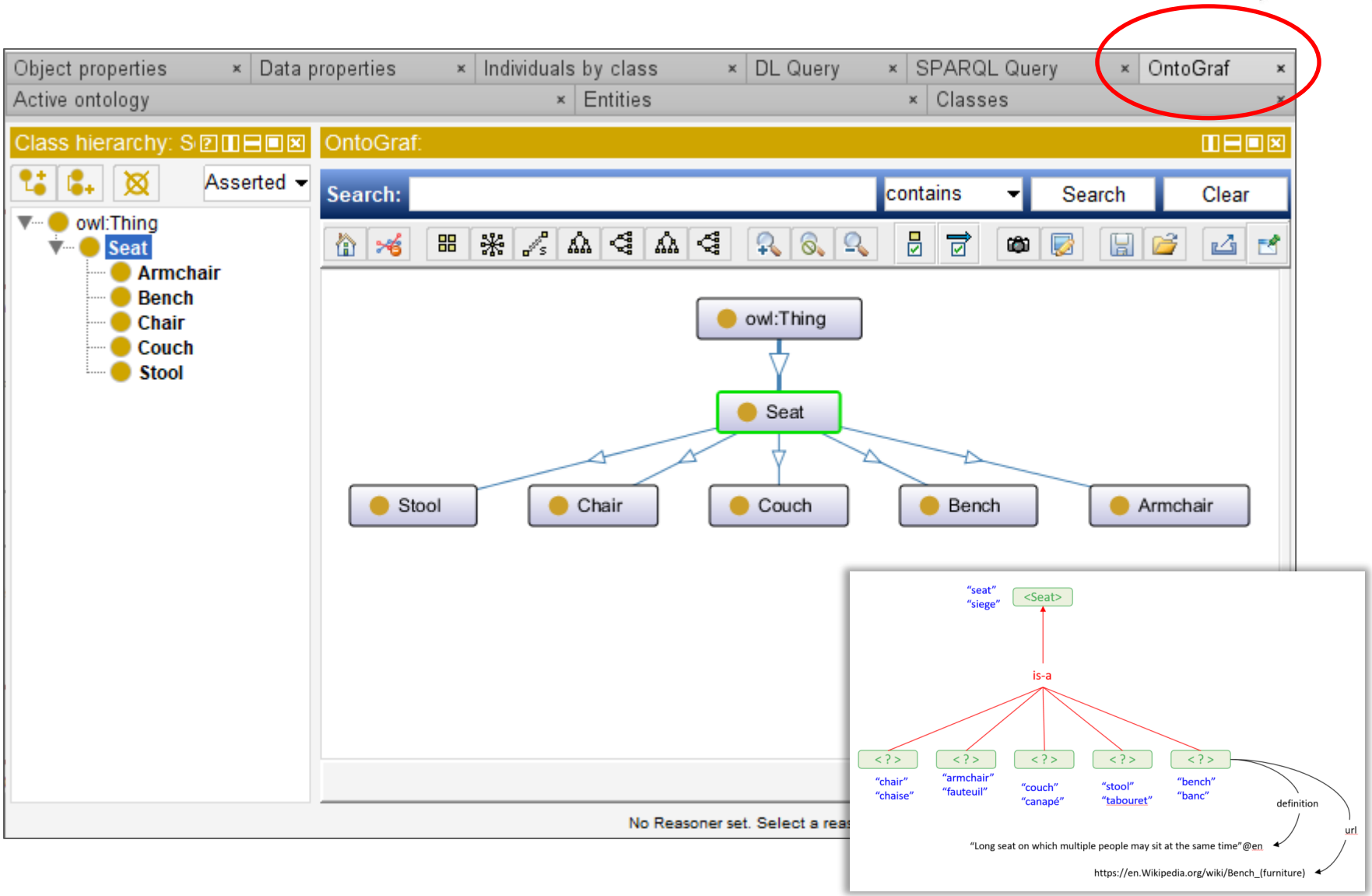
1 Defining classes 2 Organizing them 2' Adding properties



1 Defining classes 2 Organizing them 2' Adding properties



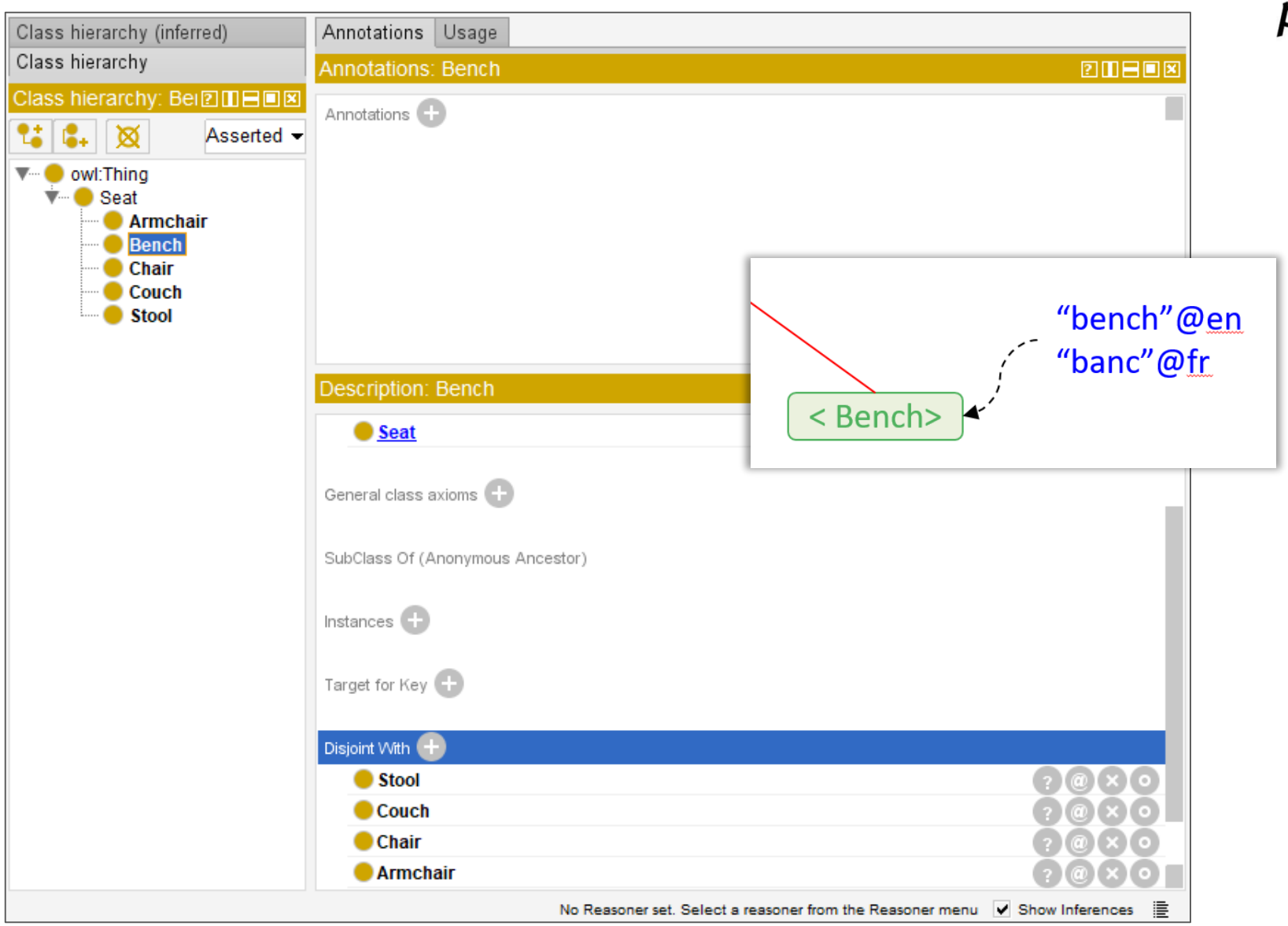
1 Defining classes 2 Organizing them 2' Adding properties



1 Defining classes

2 Organizing them

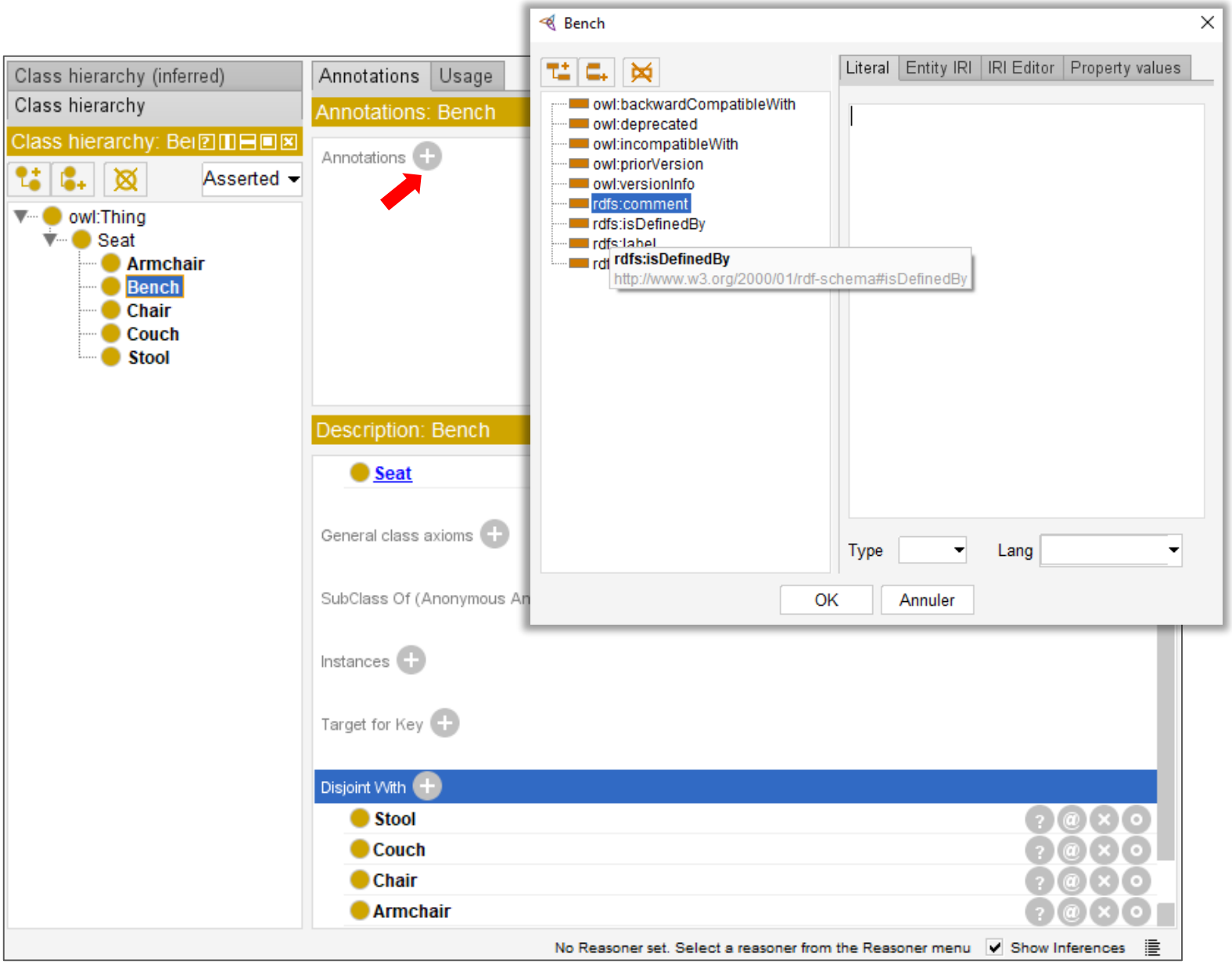
3 Annotating them



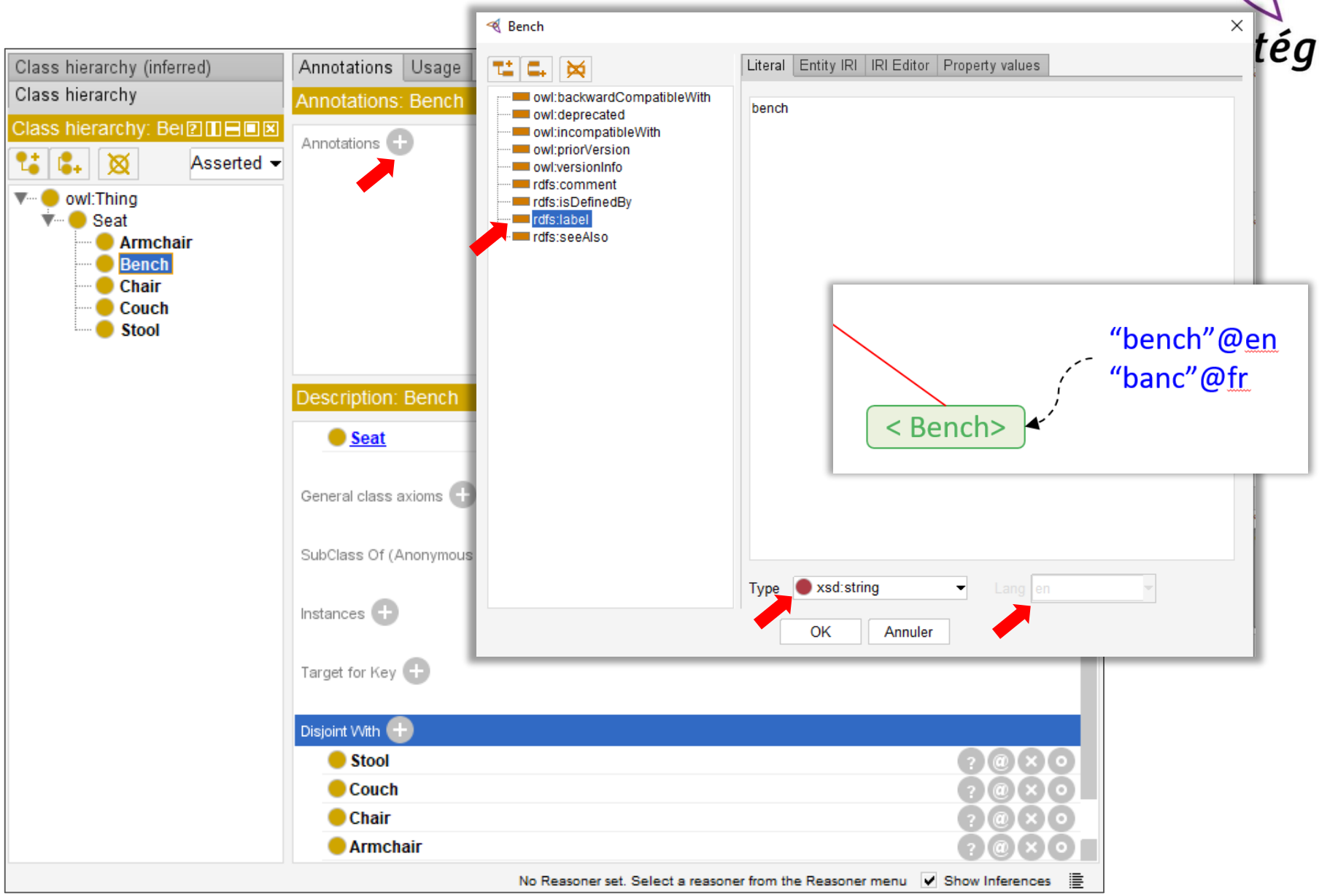
1 Defining classes

2 Organizing them

3 Annotating them



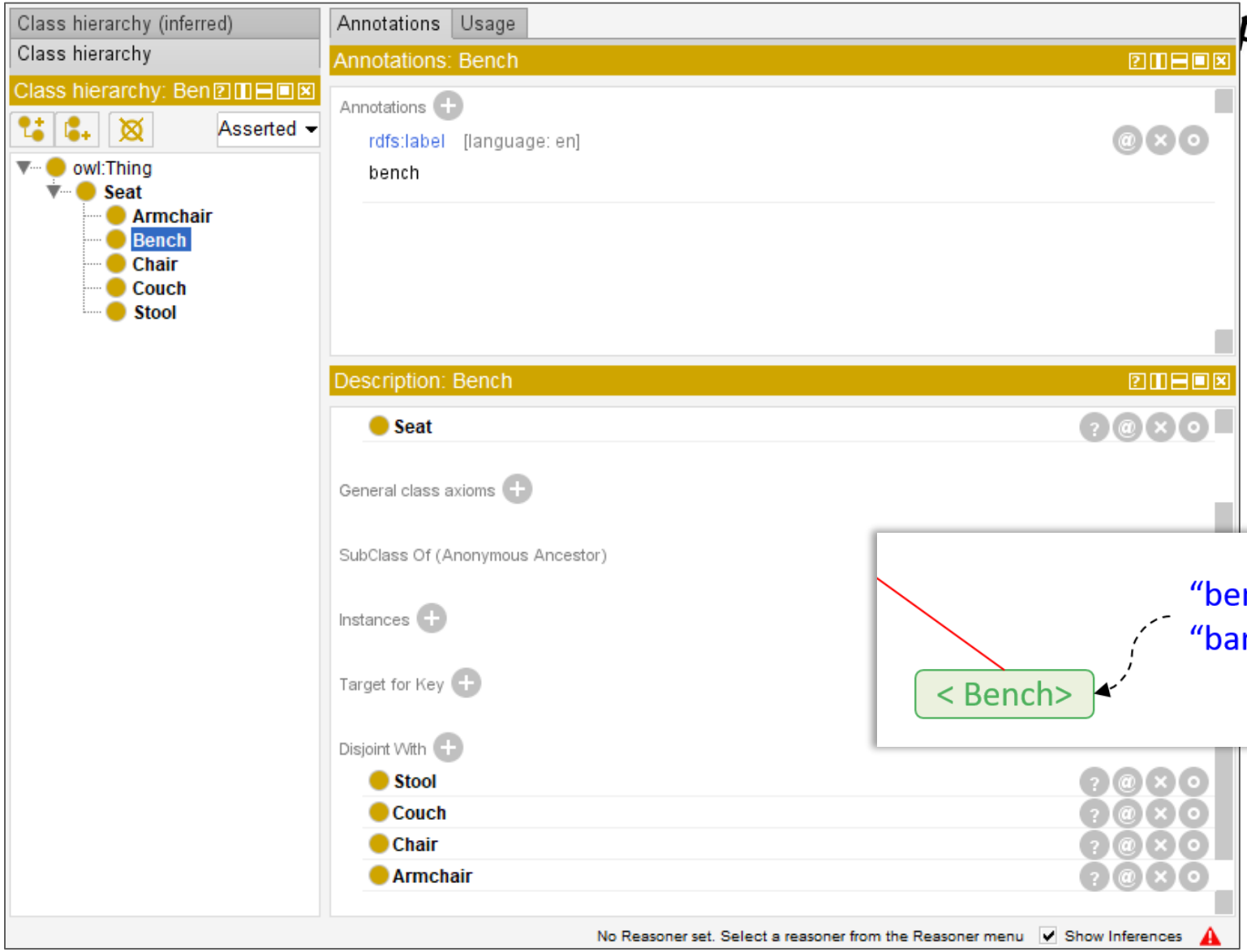
1 Defining classes 2 Organizing them 3 Annotating them



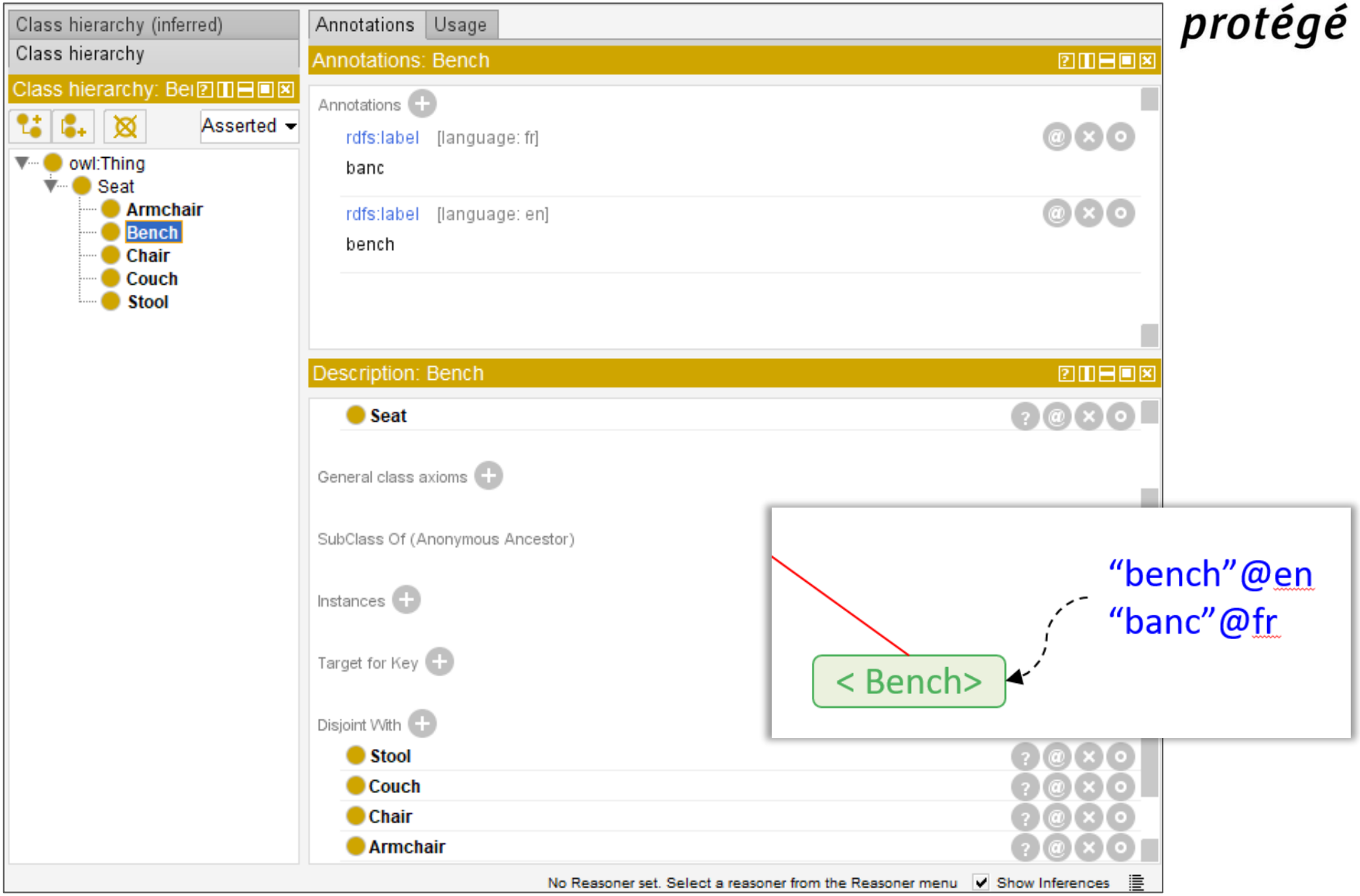
1 Defining classes

2 Organizing them

3 Annotating them



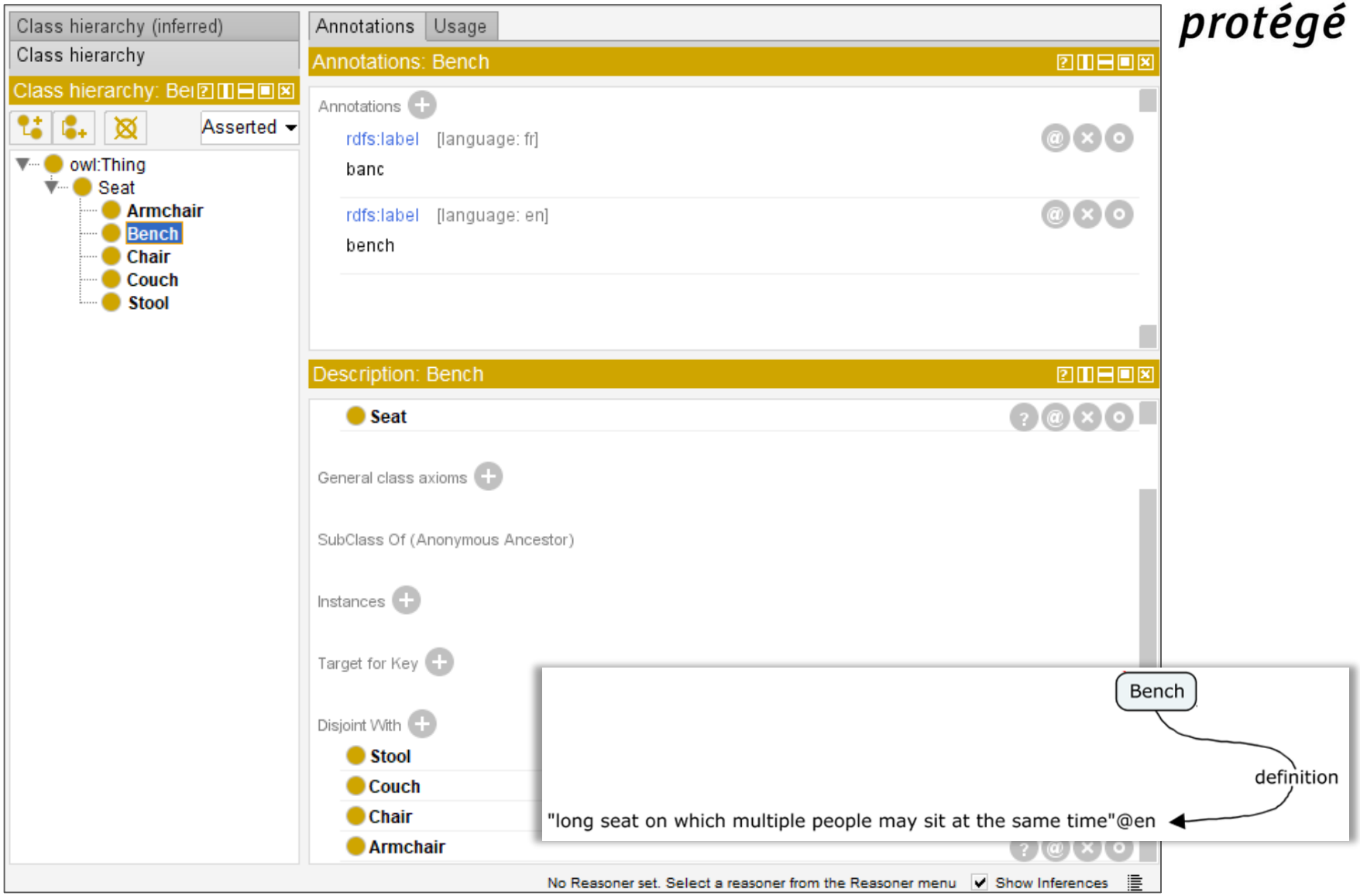
1 Defining classes 2 Organizing them 3 Annotating them



1 Defining classes

2 Organizing them

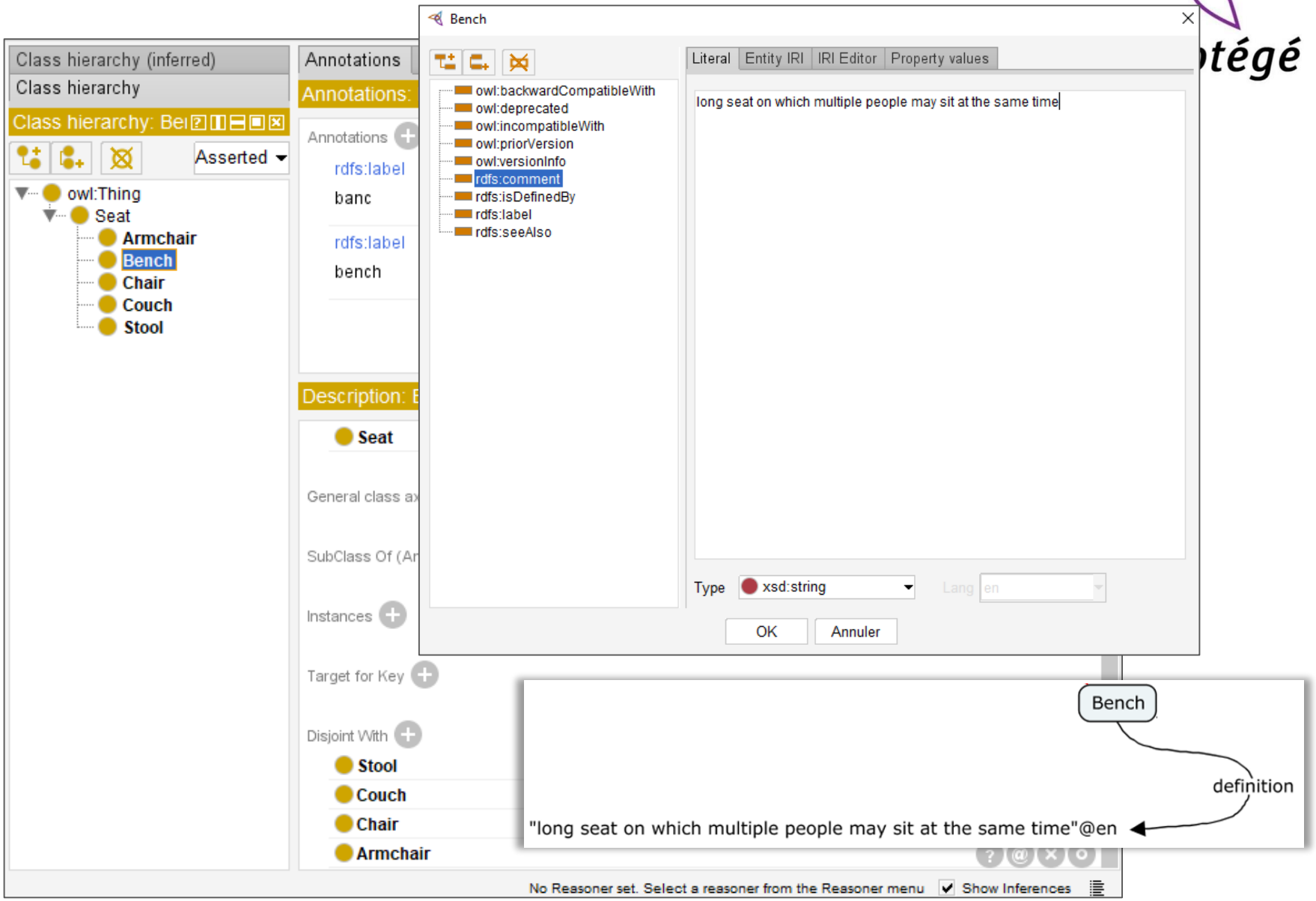
3 Annotating them



1 Defining classes

2 Organizing them

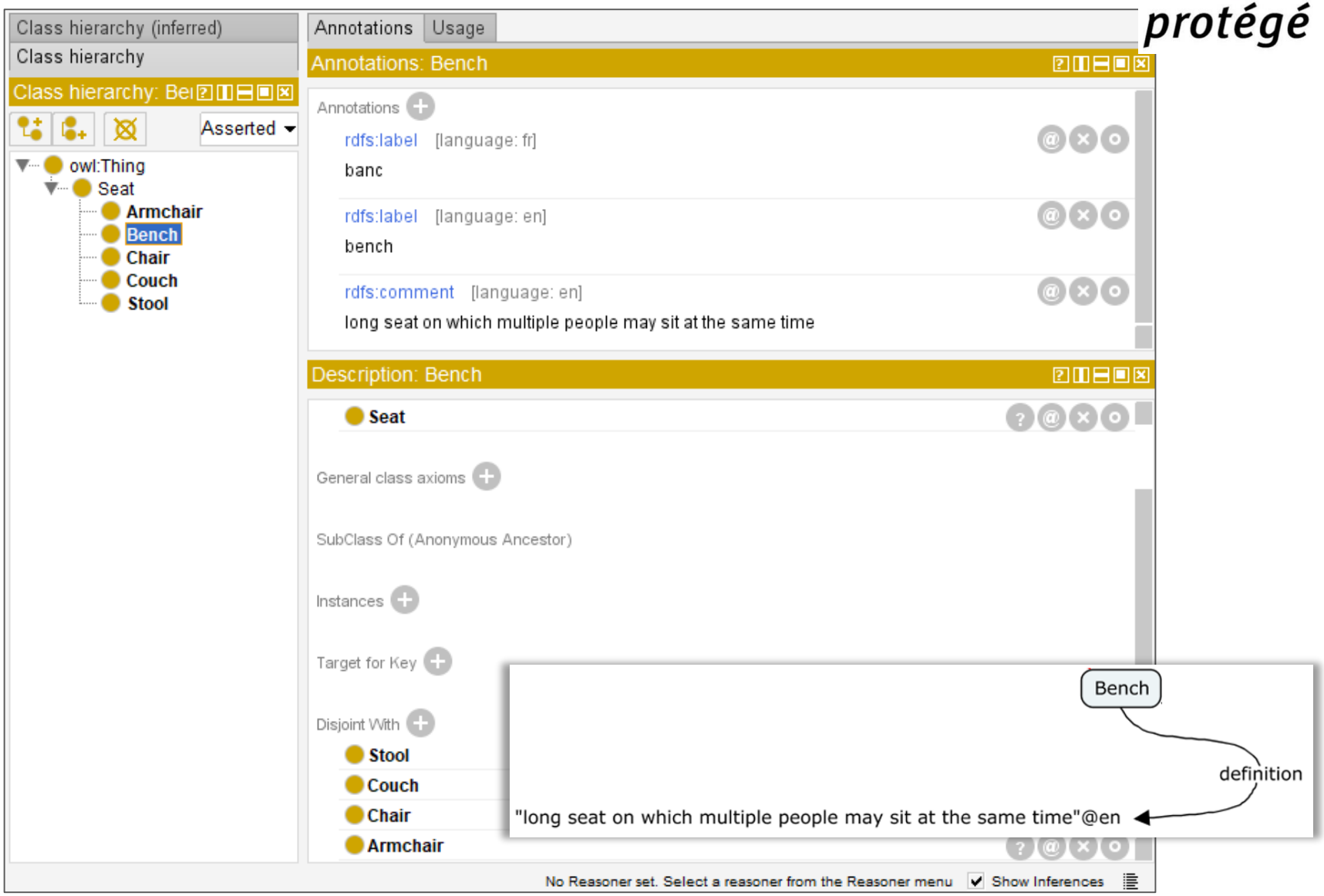
3 Annotating them



1 Defining classes

2 Organizing them

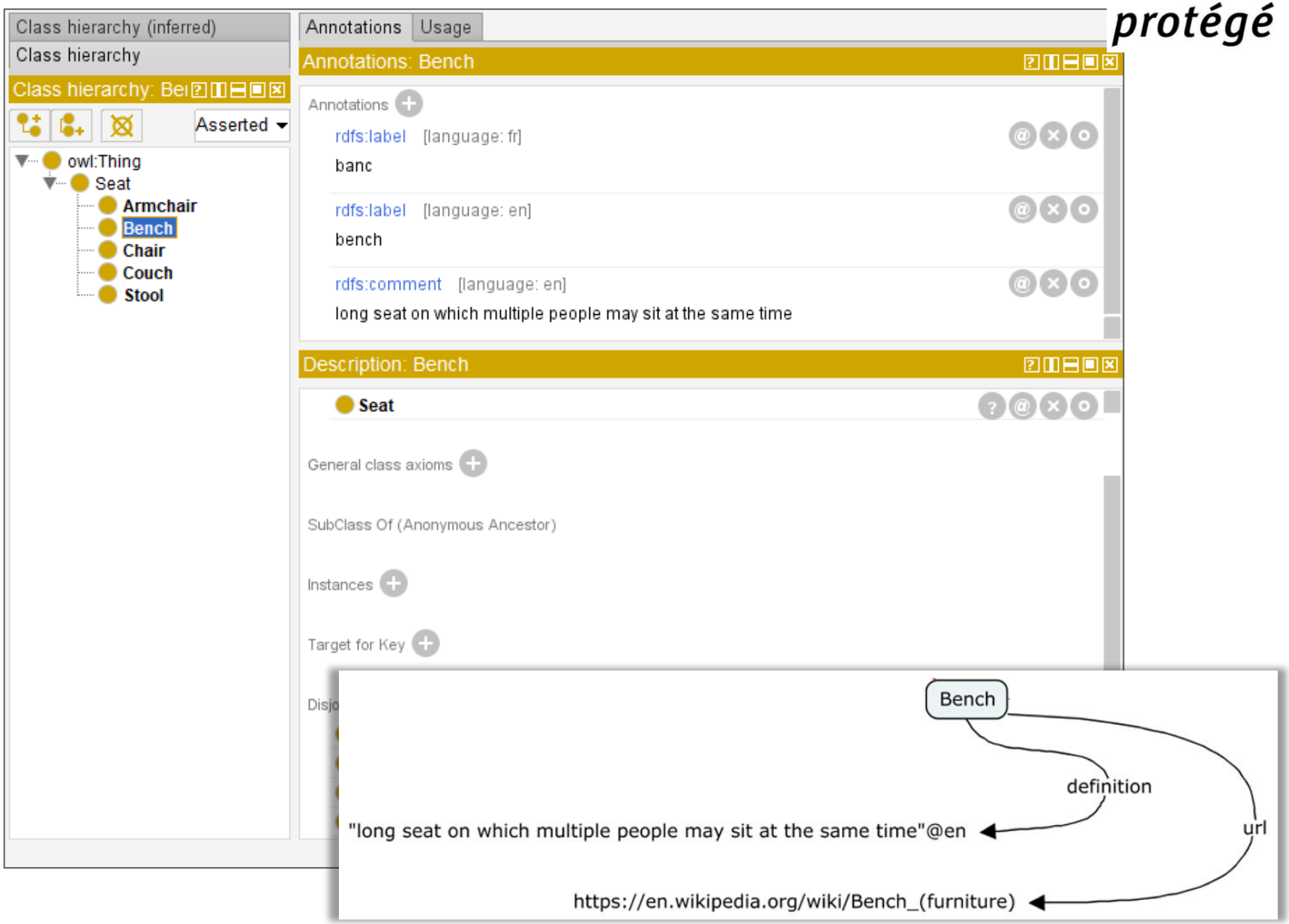
3 Annotating them



1 Defining classes

2 Organizing them

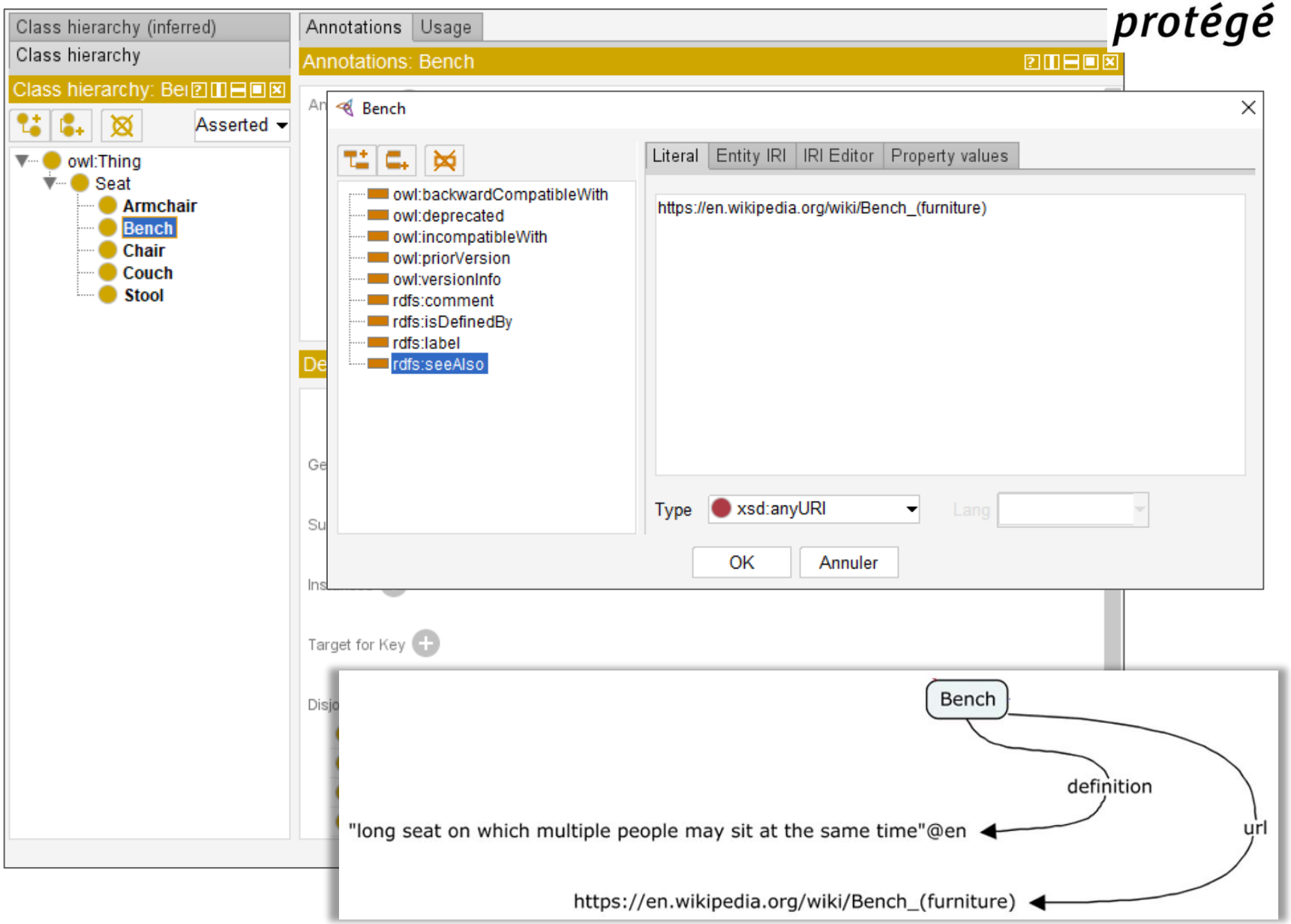
3 Annotating them



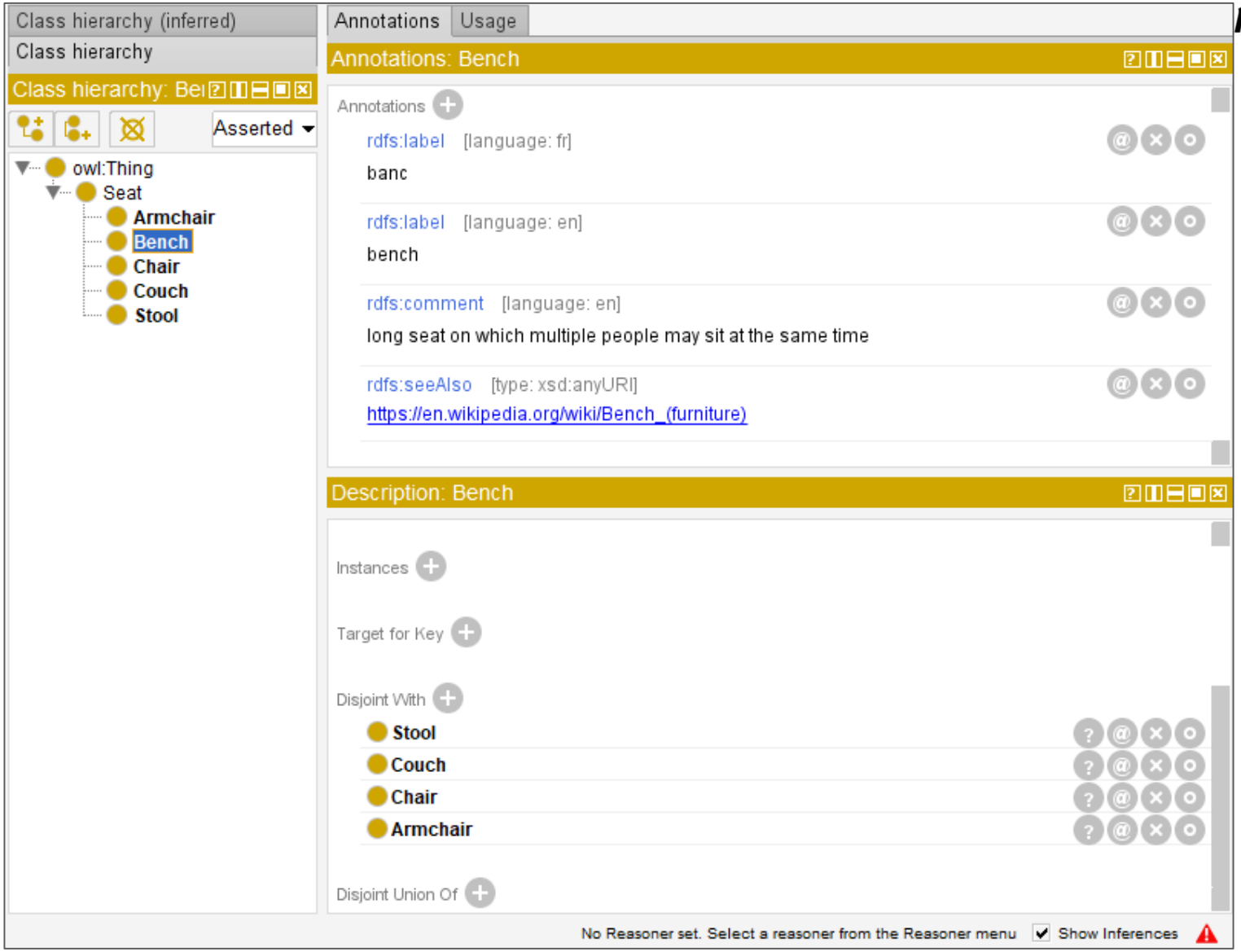
1 Defining classes

2 Organizing them

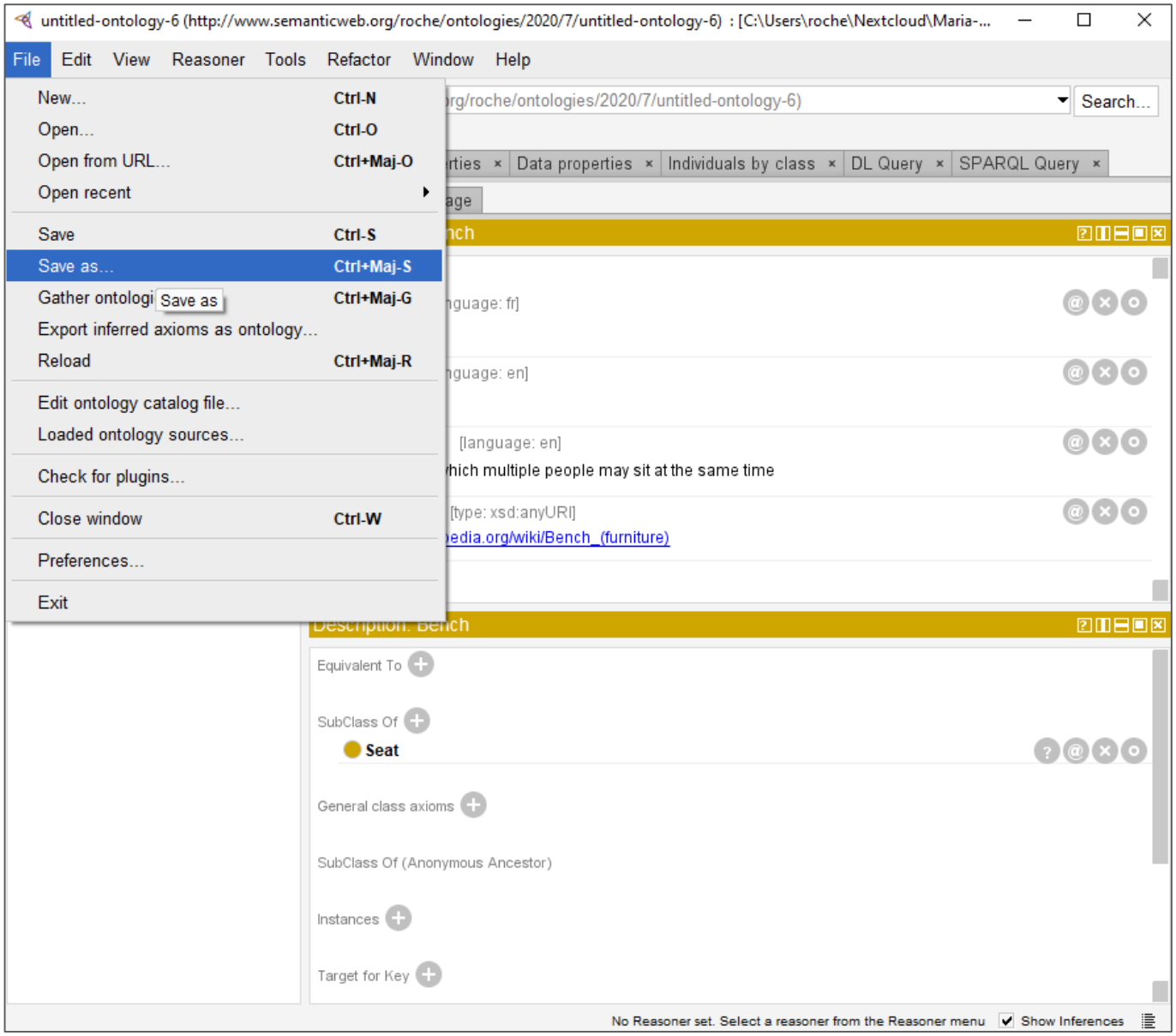
3 Annotating them



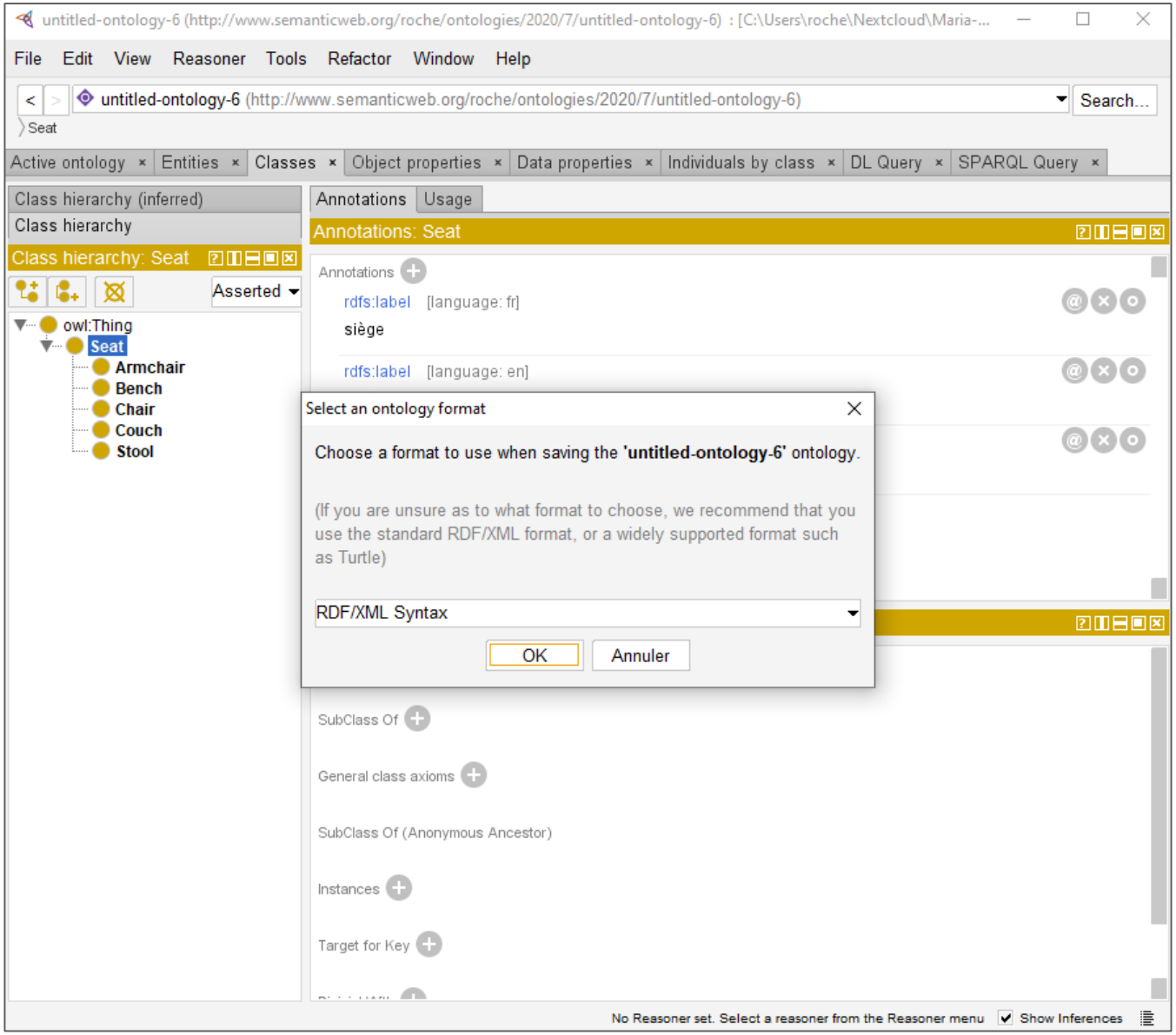
1 Defining classes 2 Organizing them 3 Annotating them



2. Querying my 1st K-Graph in Protégé



2. Querying my 1st K-Graph in Protégé



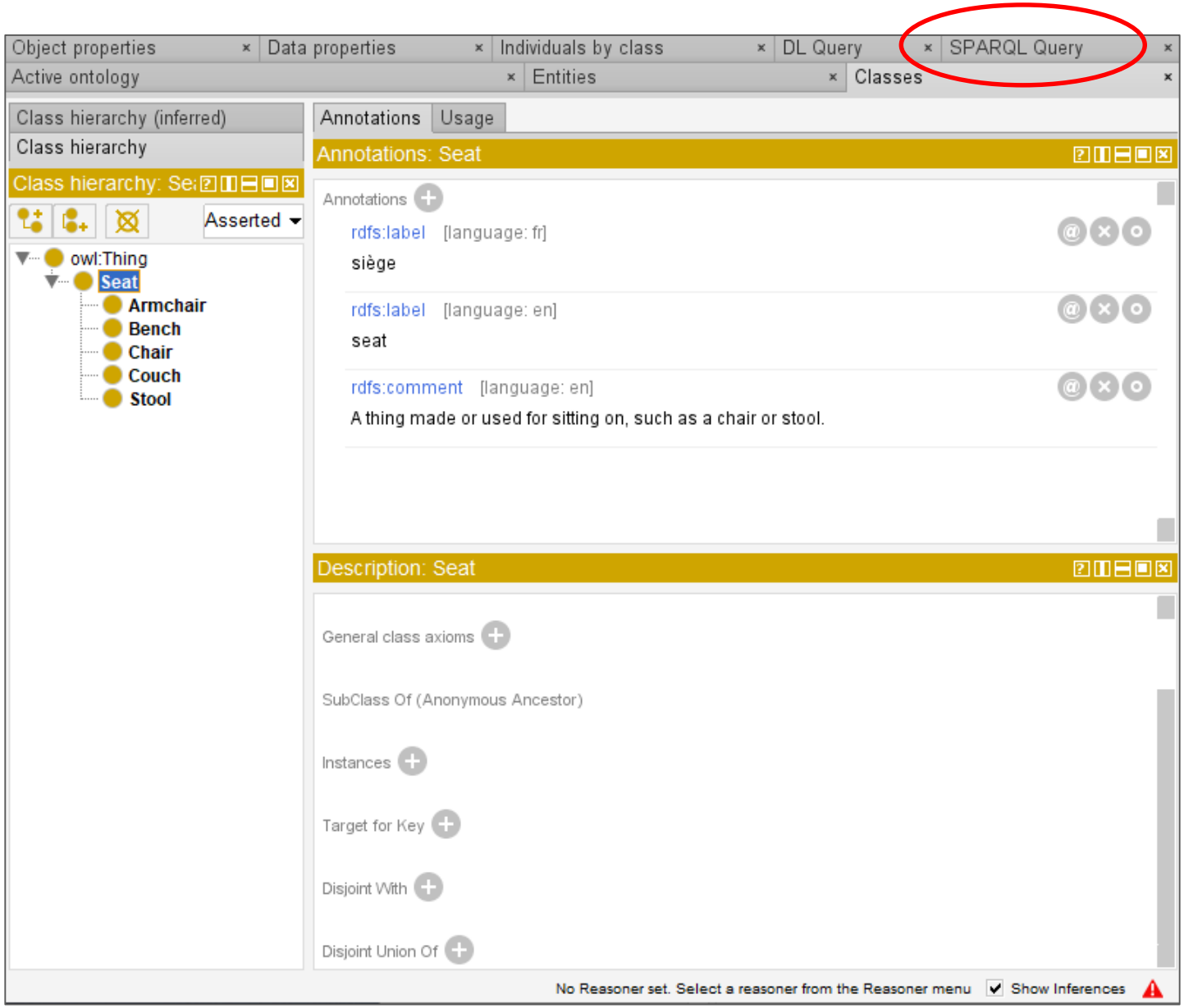
2. Querying my 1st K-Graph in Protégé

```
<?xml version="1.0"?>
<rdf:RDF xmlns="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#"
     xml:base="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6"
     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:xml="http://www.w3.org/XML/1998/namespace"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
    <owl:Ontology rdf:about="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6"/>



    <!--
    ///////////////////////////////////////////////////////////////////////////////////////
    //
    // Classes
    //
    ///////////////////////////////////////////////////////////////////////////////////////
     -->




    <!-- http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Armchair -->

    <owl:Class rdf:about="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Armchair">
        <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Seat"/>
        <owl:disjointWith rdf:resource="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Bench"/>
        <owl:disjointWith rdf:resource="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Chair"/>
```

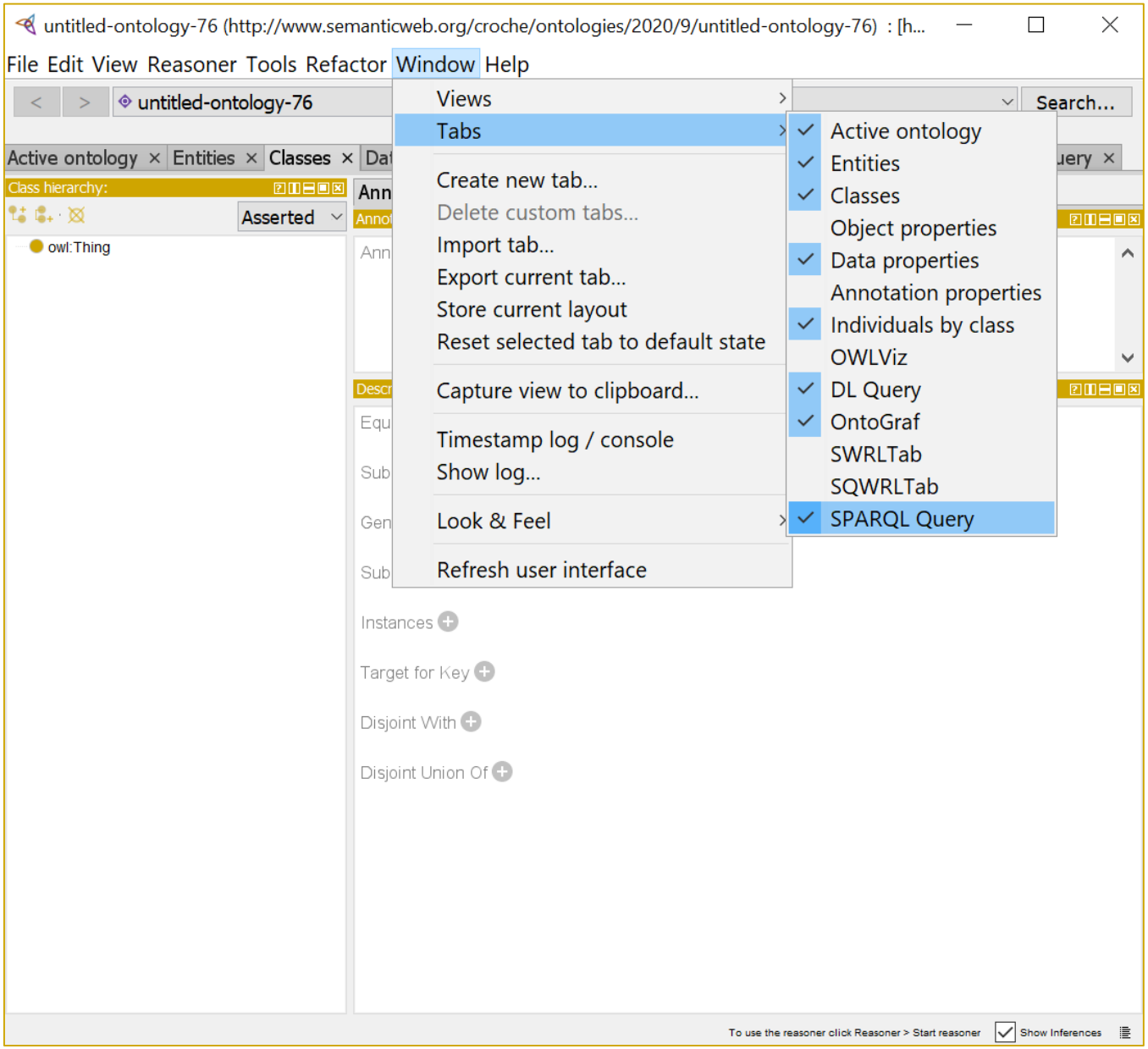
```
<!-- http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Bench -->

<owl:Class rdf:about="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Bench">
    <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Seat"/>
    <owl:disjointWith rdf:resource="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Chair"/>
    <owl:disjointWith rdf:resource="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Couch"/>
    <owl:disjointWith rdf:resource="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Stool"/>
    <rdfs:comment xml:lang="en">long seat on which multiple people may sit at the same time</rdfs:comment>
    <rdfs:label xml:lang="fr">banc</rdfs:label>
    <rdfs:label xml:lang="en">bench</rdfs:label>
    <rdfs:seeAlso rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">https://en.wikipedia.org/wiki/Bench_(furniture)</rdfs:seeAlso>
</owl:Class>
```


```
        <owl:disjointWith rdf:resource="http://www.semanticweb.org/roche/ontologies/2020/7/untitled-ontology-6#Stool"/>
        <rdfs:comment xml:lang="en">long seat on which multiple people may sit at the same time</rdfs:comment>
        <rdfs:label xml:lang="fr">banc</rdfs:label>
        <rdfs:label xml:lang="en">bench</rdfs:label>
        <rdfs:seeAlso rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">https://en.wikipedia.org/wiki/Bench_(furniture)</rdfs:seeAlso>
    </owl:Class>
```

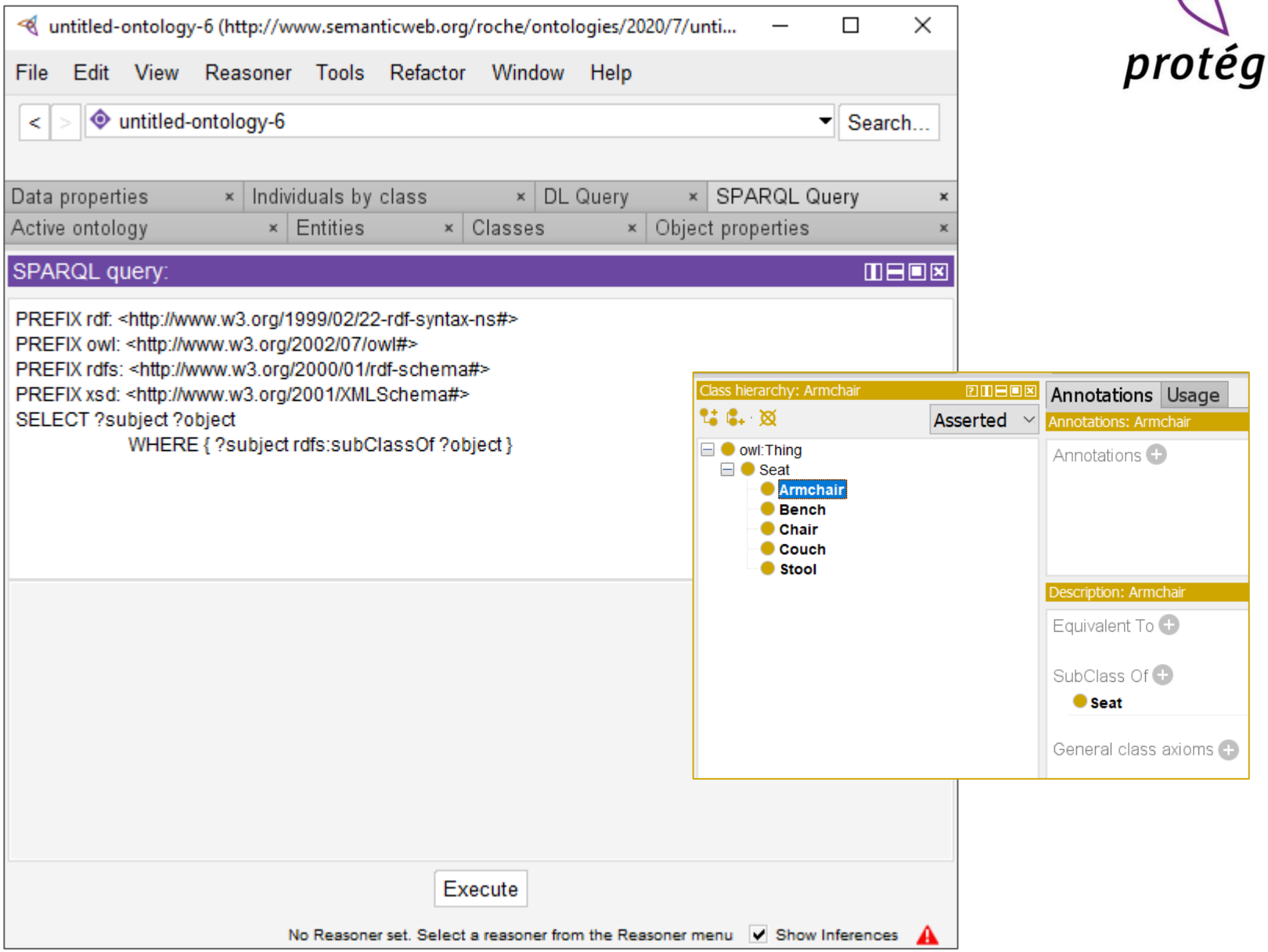
2. Querying my 1st K-Graph in Protégé



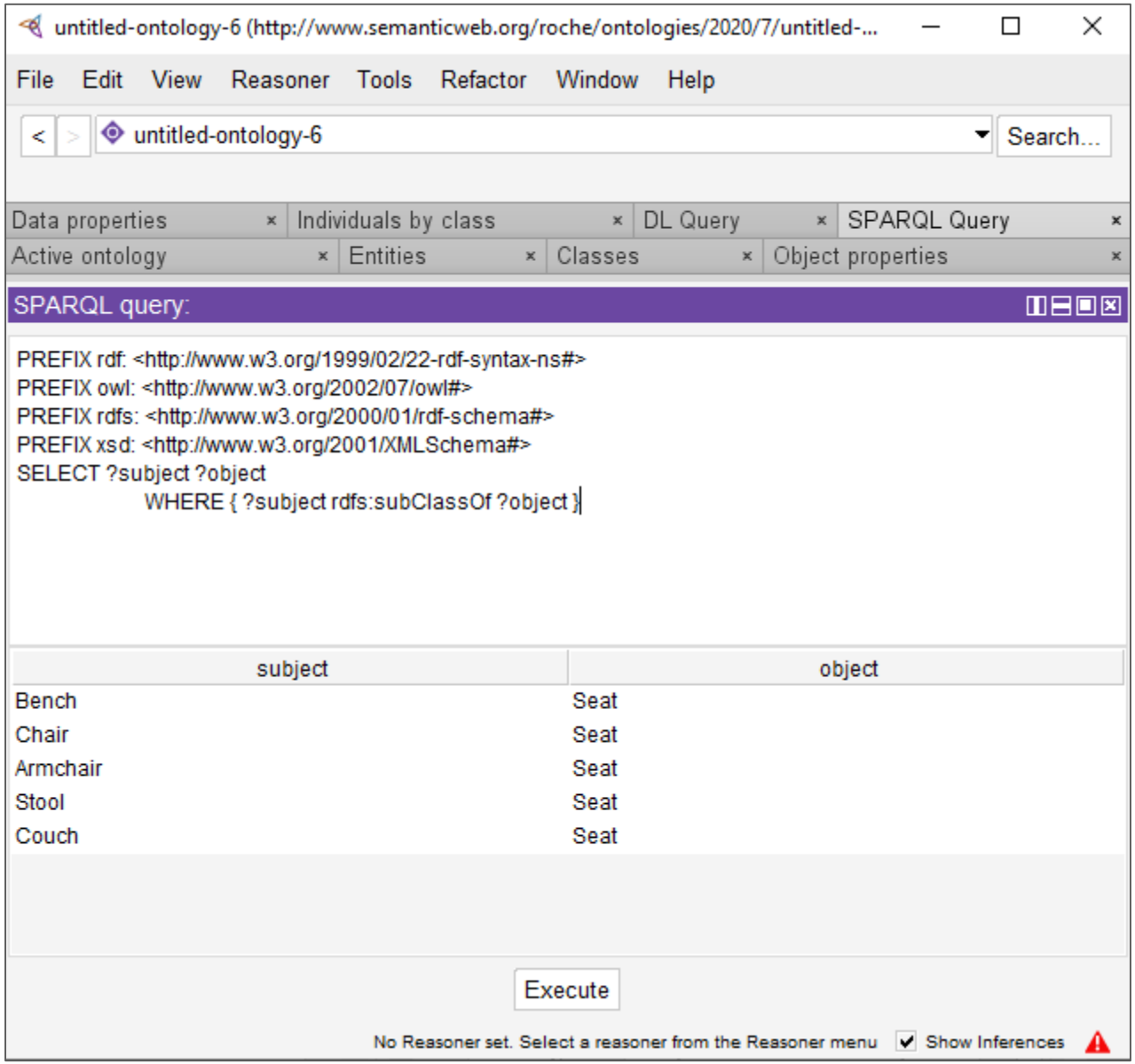
2. Querying my 1st K-Graph in Protégé



2. Querying my 1st K-Graph in Protégé

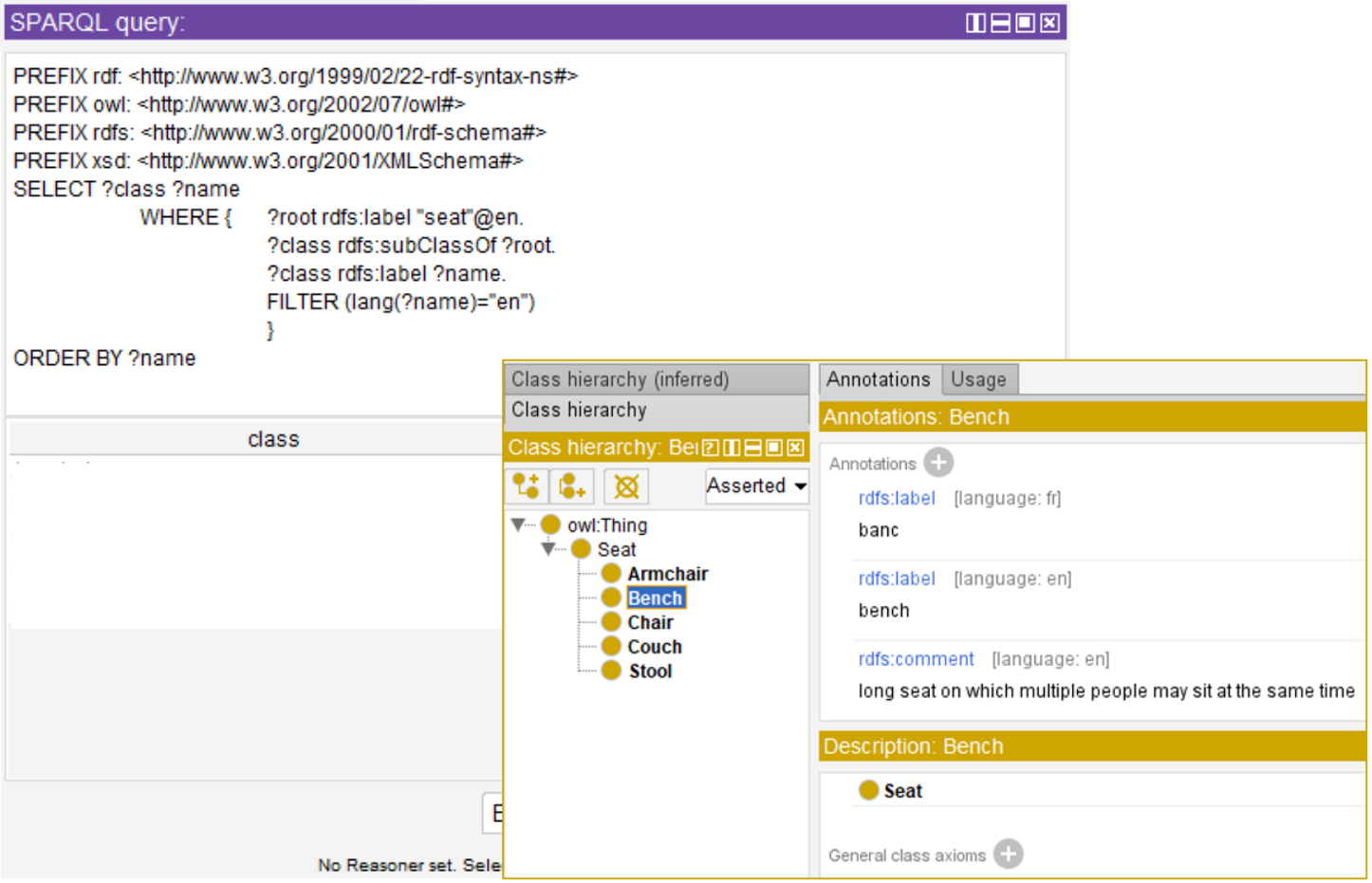


2. Querying my 1st K-Graph in Protégé



2. Querying my 1st K-Graph in Protégé

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?class ?name
                WHERE {     ?root rdfs:label "seat"@en.
                            ?class rdfs:subClassOf ?root.
                            ?class rdfs:label ?name.
                            FILTER (lang(?name)="en")
                            }
ORDER BY ?name
```



2. Querying my 1st K-Graph in Protégé

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?class ?name
                    WHERE {         ?root rdfs:label "seat"@en.
                                    ?class rdfs:subClassOf ?root.
                                    ?class rdfs:label ?name.
                                    FILTER (lang(?name)="en")
                                    }
ORDER BY ?name
```

SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?class ?name
                    WHERE {     ?root rdfs:label "seat"@en.
                                ?class rdfs:subClassOf ?root.
                                ?class rdfs:label ?name.
                                FILTER (lang(?name)="en")
                                }
ORDER BY ?name
```

class	name
Armchair	"armchair"@en
Bench	"bench"@en
Chair	"chair"@en
Couch	"couch"@en
Stool	"stool"@en

Execute

No Reasoner set. Select a reasoner from the Reasoner menu ☑ Show Inferences

2. Querying my 1st K-Graph in Protégé

SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?name_en ?name_fr
        WHERE {     ?subject rdfs:label ?name_en.
                    FILTER (lang(?name_en)="en").
                    ?subject rdfs:label ?name_fr.
                    FILTER (lang(?name_fr)="fr") }
ORDER BY ?name_en
```

name_en	name_fr

No Reasoner set. Selec

Class hierarchy (inferred)

Class hierarchy

Class hierarchy: Bei

Asserted

- owl:Thing
  - Seat
    - Armchair
    - Bench
    - Chair
    - Couch
    - Stool

Annotations | Usage

Annotations: Bench

Annotations

rdfs:label [language: fr]
banc

rdfs:label [language: en]
bench

rdfs:comment [language: en]
long seat on which multiple people may sit at the same time

Description: Bench

Seat

General class axioms

2. Querying my 1st K-Graph in Protégé

SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?name_en ?name_fr
        WHERE {   ?subject rdfs:label ?name_en.
                  FILTER (lang(?name_en)="en").
                  ?subject rdfs:label ?name_fr.
                  FILTER (lang(?name_fr)="fr") }
ORDER BY ?name_en
```

name_en	name_fr
"armchair"@en	"fauteuil"@fr
"bench"@en	"banc"@fr
"chair"@en	"chaise"@fr
"couch"@en	"canapé"@fr
"seat"@en	"siège"@fr
"stool"@en	"tabouret"@fr

Execute

No Reasoner set. Select a reasoner from the Reasoner menu ☑ Show Inferences

2. Querying my 1st K-Graph in Protégé

Display the English terms of seat with their definition in English

Draw the query graph

2. Querying my 1st K-Graph in Protégé

Display the English terms of seat with their definition in English

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT  ?name ?definition
     WHERE {     ?root rdfs:label "seat"@en.
                 ?class rdfs:subClassOf ?root.
                 ?class rdfs:label ?name.
                 ?class rdfs:comment ?definition.
                 FILTER (lang(?name)="en")
}
ORDER BY ?name
```

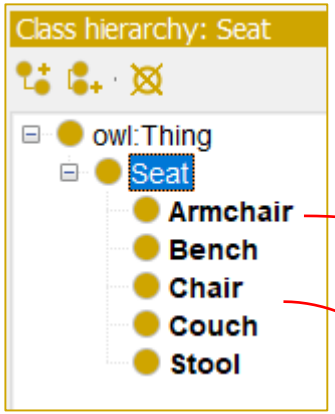
SPARQL query:

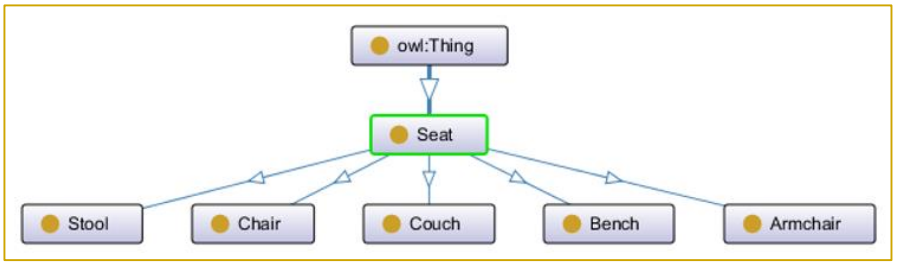
```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT  ?name ?definition
     WHERE {     ?root rdfs:label "seat"@en.
                 ?class rdfs:subClassOf ?root.
                 ?class rdfs:label ?name.
                 ?class rdfs:comment ?definition.
                 FILTER (lang(?name)="en")
                 }
ORDER BY ?name
```

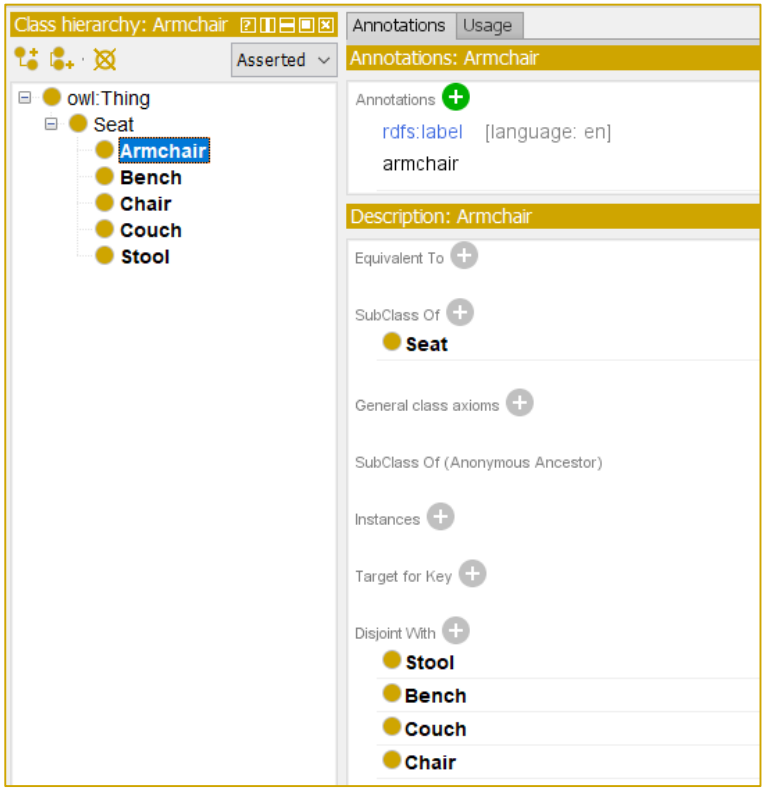
name	definition
"armchair"@en	"Seat for one person wiht feet and back with arms"@en
"bench"@en	"Seat for several persons wiht feet without back and without arms"@en
"chair"@en	"chaise"@fr
"chair"@en	"Seat for one person wiht feet and back without arms"@en
"couch"@en	"Seat for several persons wiht feet, back and arms"@en
"stool"@en	"Seat for one person wiht feet without back without arms"@en

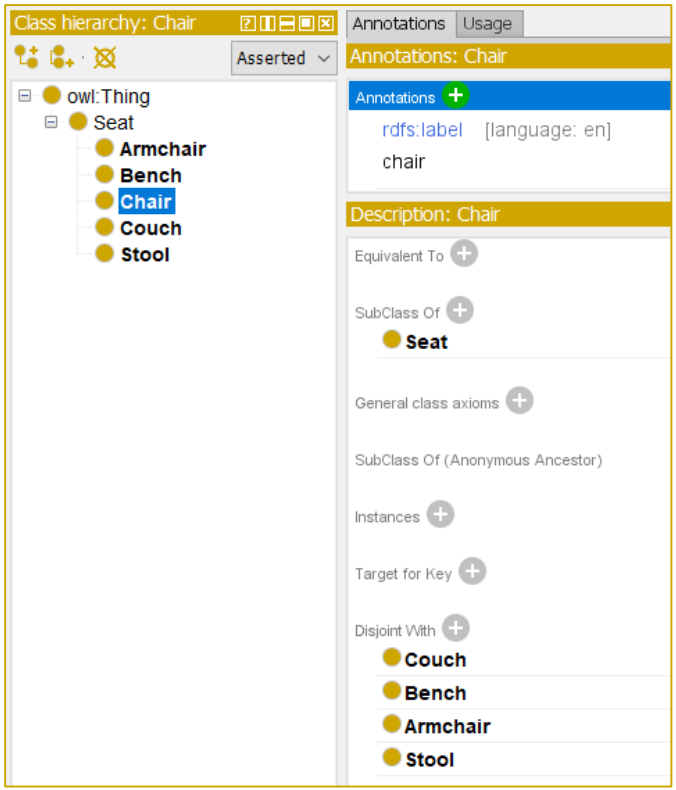
We need go a step further

A Step Further

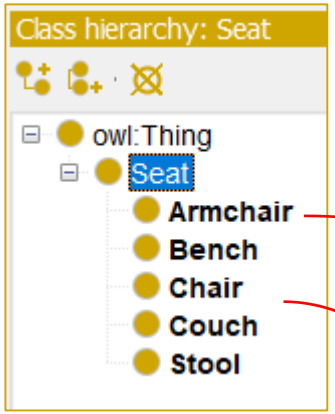


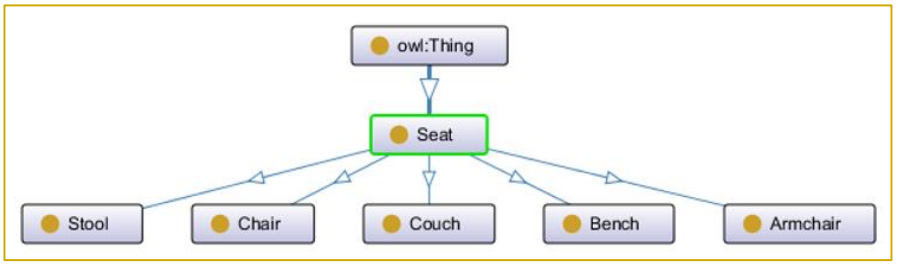


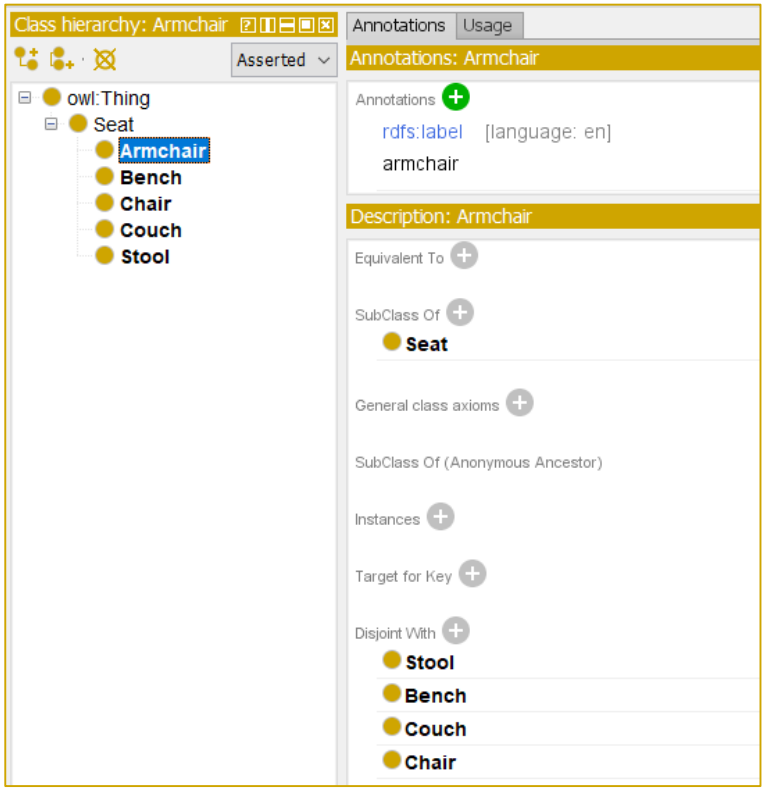


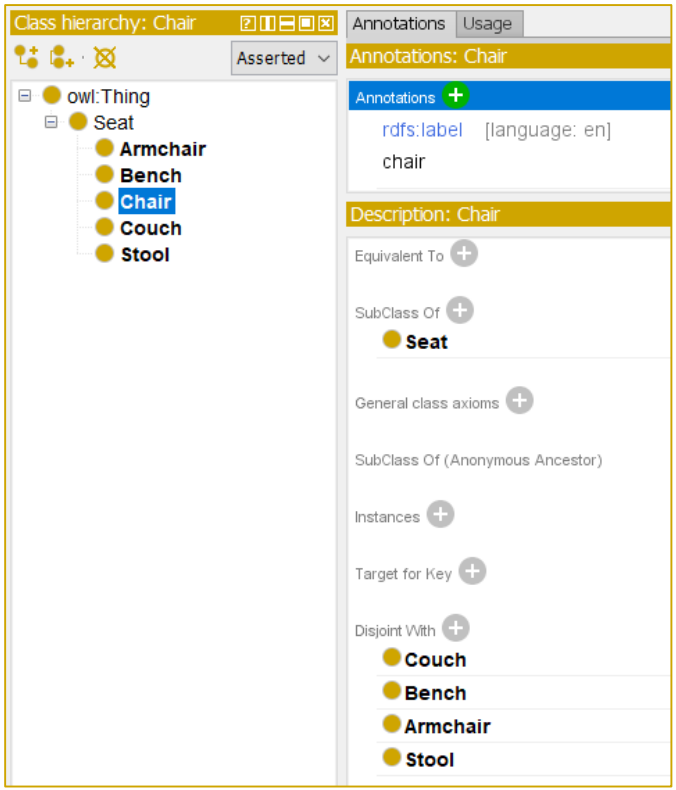


A Step Further









What is the difference between Chair and Armchair, Bench, Couch, Stool?

What is their definition?

A Step Further

Classes

```
<owl:Class rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Chair">
    <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Seat"/>
    <rdfs:comment xml:lang="en">Seat for one person wiht feet and back without arms</rdfs:comment>
    <rdfs:comment xml:lang="fr">chaise</rdfs:comment>
    <rdfs:label xml:lang="en">chair</rdfs:label>
  </owl:Class>
```

```
<owl:Class rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Armchair">
    <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Seat"/>
    <rdfs:comment xml:lang="en">Seat for one person wiht feet and back with arms</rdfs:comment>
    <rdfs:label xml:lang="en">armchair</rdfs:label>
    <rdfs:label xml:lang="fr">fauteuil</rdfs:label>
  </owl:Class>
```

General axioms

```
<rdf:Description>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AllDisjointClasses"/>
    <owl:members rdf:parseType="Collection">
      <rdf:Description rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Armchair"/>
      <rdf:Description rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Bench"/>
      <rdf:Description rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Chair"/>
      <rdf:Description rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Couch"/>
      <rdf:Description rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Stool"/>
    </owl:members>
  </rdf:Description>
```

A Step Further

Classes

```
<owl:Class rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Chair">
    <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Seat"/>
    <rdfs:comment xml:lang="en">Seat for one person wiht feet and back without arms</rdfs:comment>
    <rdfs:comment xml:lang="fr">chaise</rdfs:comment>
    <rdfs:label xml:lang="en">chair</rdfs:label>
  </owl:Class>
```

```
<owl:Class rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Armchair">
    <rdfs:subClassOf rdf:resource="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Seat"/>
    <rdfs:comment xml:lang="en">Seat for one person wiht feet and back with arms</rdfs:comment>
    <rdfs:label xml:lang="en">armchair</rdfs:label>
    <rdfs:label xml:lang="fr">fauteuil</rdfs:label>
  </owl:Class>
```

General axioms

```
<rdf:Description>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AllDisjointClasses"/>
    <owl:members rdf:parseType="Collection">
      <rdf:Description rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Armchair"/>
      <rdf:Description rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Bench"/>
      <rdf:Description rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Chair"/>
      <rdf:Description rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Couch"/>
      <rdf:Description rdf:about="http://www.semanticweb.org/croche/ontologies/2020/4/untitled-ontology-22#Stool"/>
    </owl:members>
  </rdf:Description>
```

Class hierarchy: Bag

Asserted

- owl:Thing
  - 311ROThing
    - 311Type
    - Action
    - Agency
    - LocationType
    - MessageCategory
    - ServiceRequest
    - SPSPoint
    - Subject
      - GarbageContainer
        - Bag
        - Basket
        - Bin
        - Box
        - Cardboard
        - Cart
        - Pile
        - Waste
      - Pest
      - Plants
      - RoadSymbol
      - Sign
      - Vehicle
  - Feature
  - foaf:Person

Annotations | Usage

Annotations: Bag

Annotations

rdfs:label [language: en]

Bag

Description: Bag

Equivalent To

SubClass Of

GarbageContainer

General class axioms

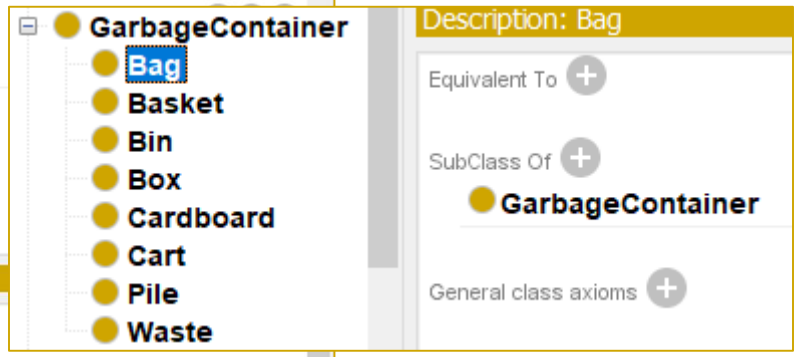
SubClass Of (Anonymous Ancestor)

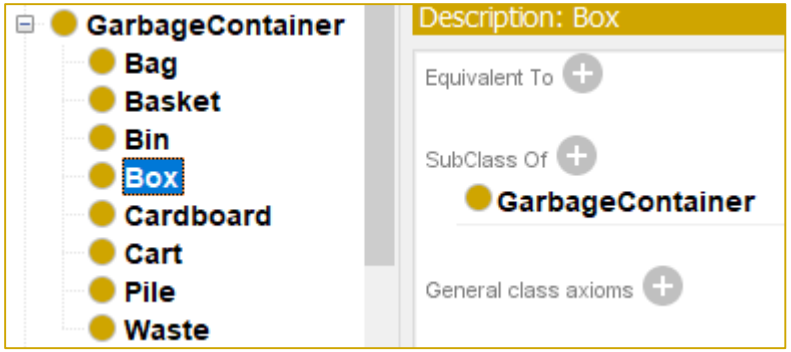
Instances

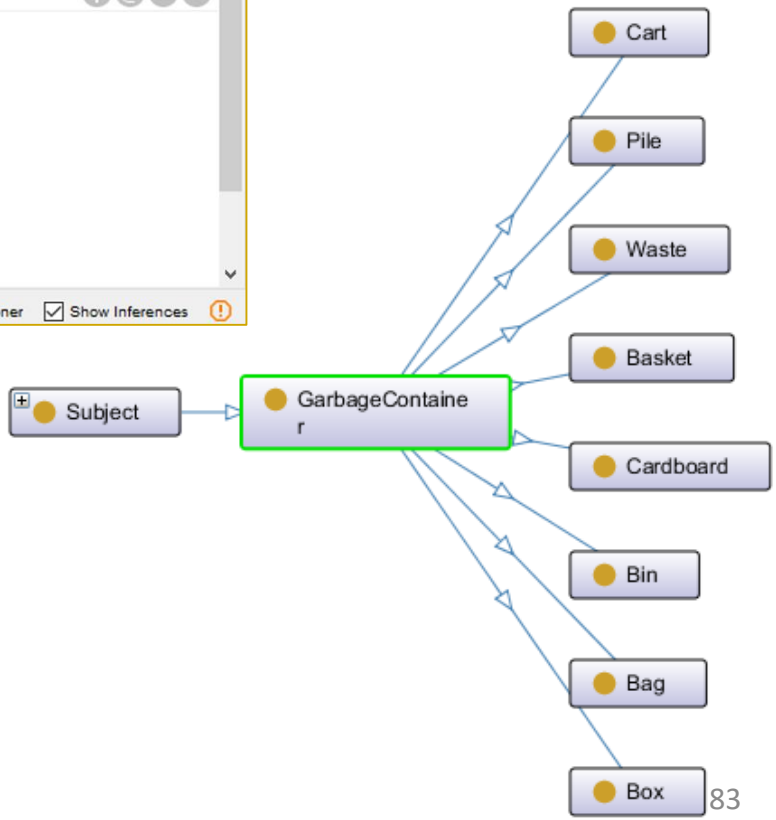
Target for Key

To use the reasoner click Reasoner > Start reasoner

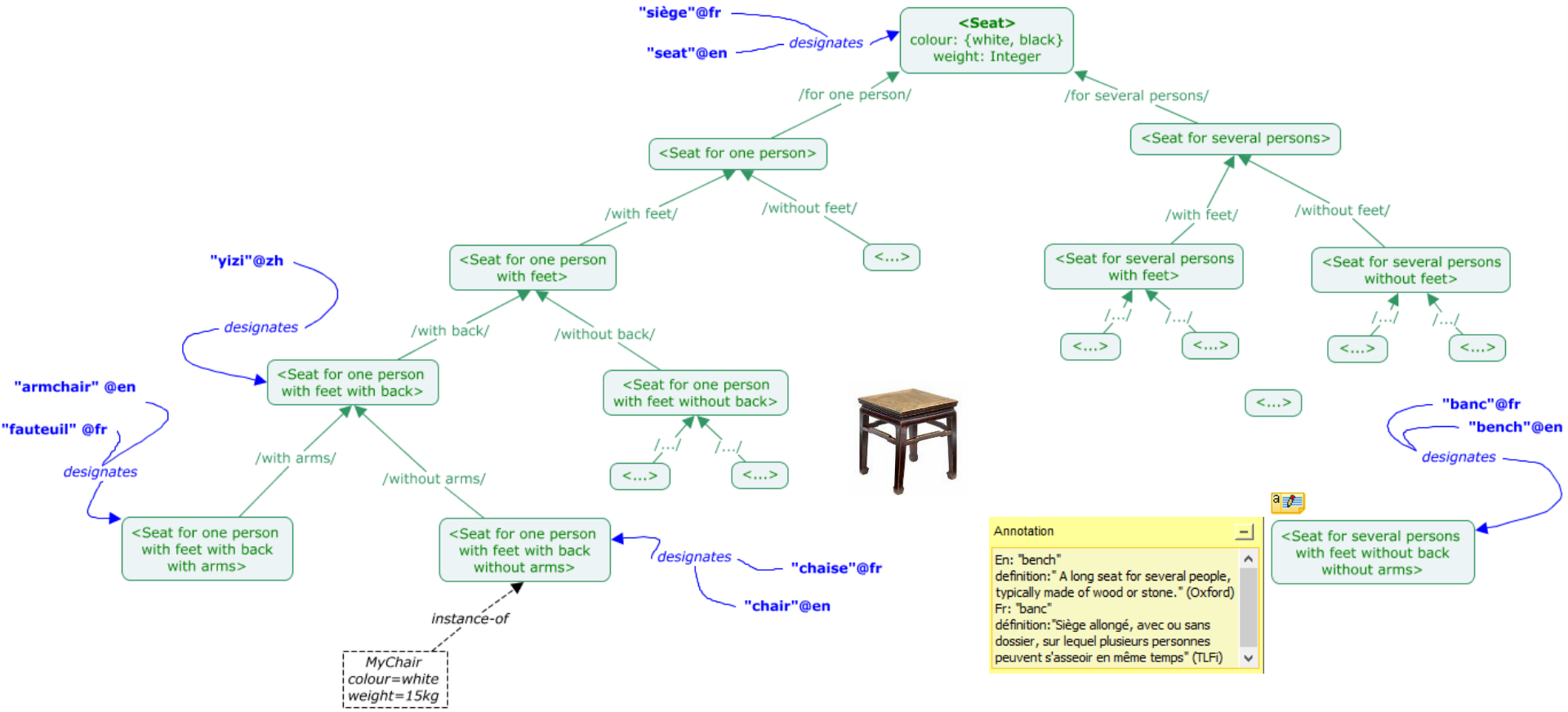
Show Inferences







Ontoterminology of Seats



Array of differences

Objects	Concepts	Axis of analysis		Axis of analysis		Axis of analysis		Axis of analysis		Terms	
Objects	Concepts	for one person	several persons	with feet	without feet	with back	without back	with arms	without arms	Designations (English)	Designations (French)
	<Seat 1 person with feet with back without arms>	X		X		X			X	"chair"	"chaise"
	<Seat 1 person with feet with back with arms>	X		X		X		X		"armchair"	"fauteuil"
	<Seat 1 person with feet without back without arms>	X		X			X		X	"stool"	"tabouret"
	<Seat several persons with feet with back with arms>		X	X		X		X		"couch"	"canapé"
	<Seat several persons with feet without back without arms>		X	X			X		X	"bench"	"banc"

"chair" : Seat for one person with feet and back without arms.

<Seat for one person with feet with back without arms>

::= <Seat> + /for one person/ + /with feet/ + /with back/ + /without arms/

"armchair" : Seat for one person with feet and back with arms.

<Seat for one person with feet with back without arms>

::= <Seat> + /for one person/ + /with feet/ + /with back/ + /with arms/

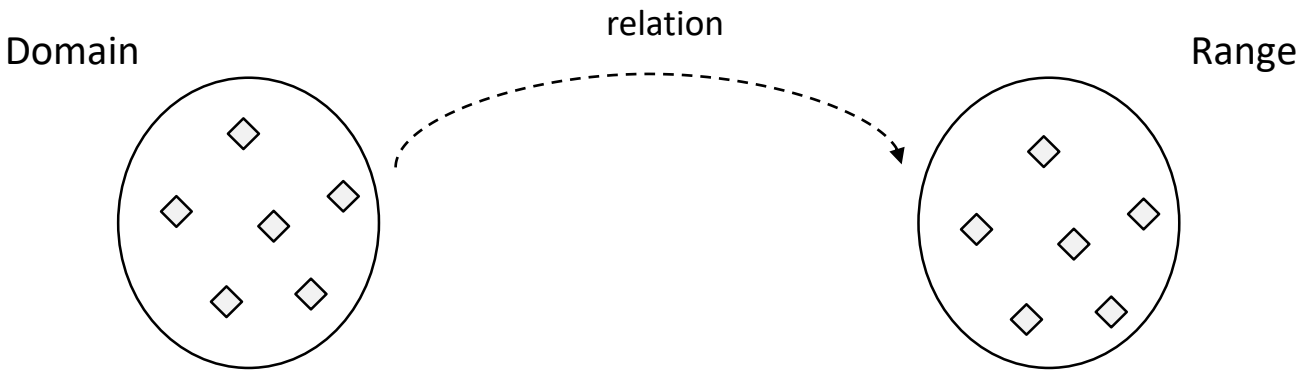
"bench" : Seat for several persons with feet, without back, and without arms.

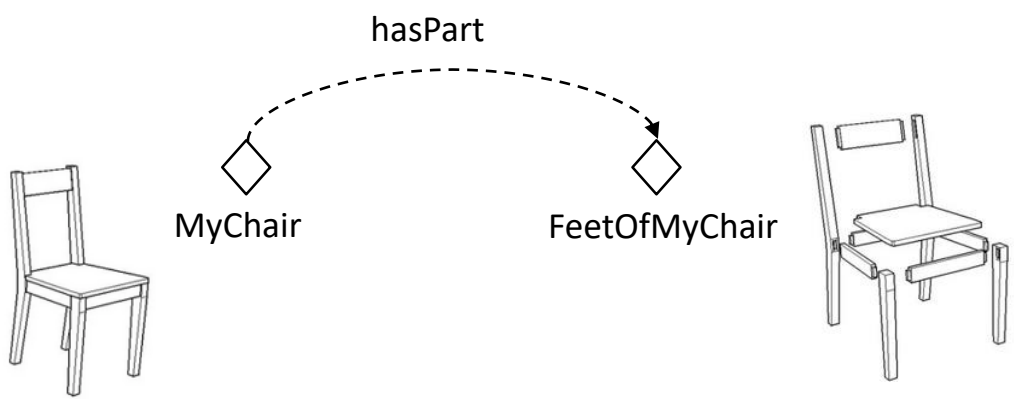
<Seat for one person with feet with back without arms>

::= <Seat> + /for one person/ + /with feet/ + /with back/ + /with arms/

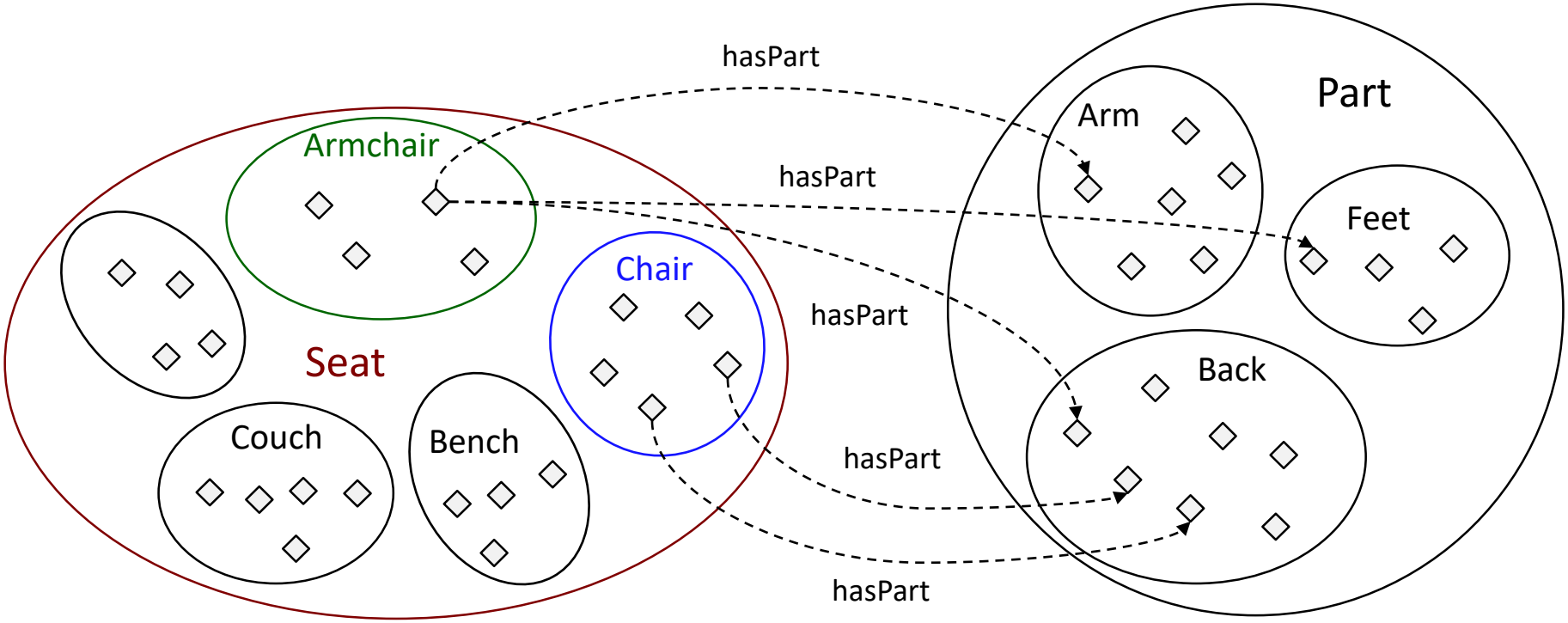
Building an OWL Ontology using Protégé: Object Properties

Relationships between individuals





Building an OWL Ontology using Protégé: Object Properties



Building an OWL Ontology using Protégé: Defining Classes

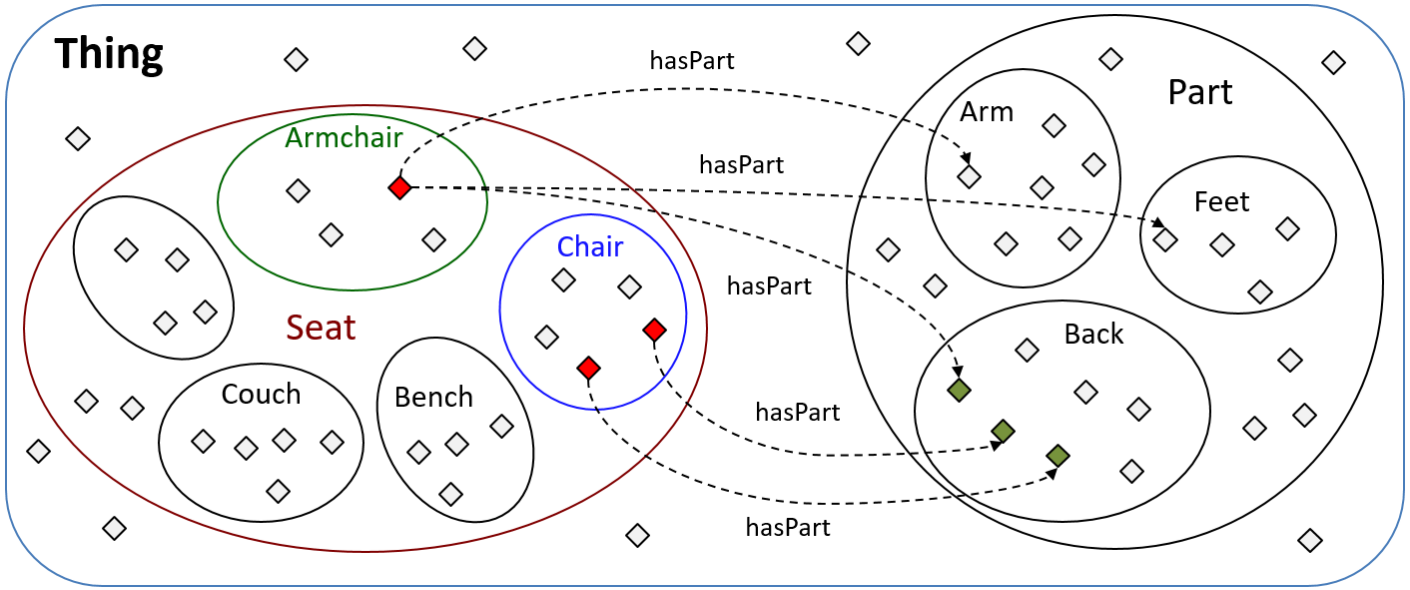
(Object or Data) **Properties Restrictions** can be used to define classes

The key idea is that a class of individuals is described or defined by the relationships that these individuals participate in

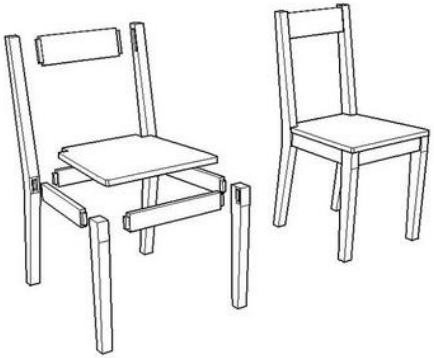
A **restriction** describes an **anonymous class** (an unnamed class).

The anonymous class contains all of the individuals that satisfy the restriction (i.e. all of the individuals that have the relationships required to be a member of the class).

SeatWithBack = { x / ∃ y Back(y) ∧ hasPart(x,y) }

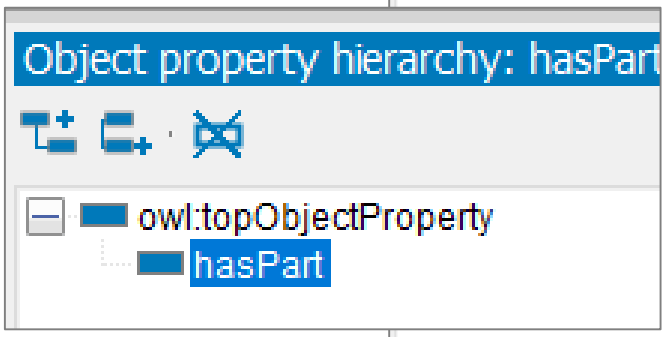


Building an OWL Ontology using Protégé: Object Properties



hasPart

MyChair FeetOfMyChair



untitled-ontology-22 (http://www.semanticweb.org/croche/ontologies/2020/4/u...

File Edit View Reasoner Tools Refactor Window Help

< > untitled-ontology-22 Search...

hasPart

Object properties | Individuals by class | OntoGraf | SPARQL Query | Data properties

Active ontology | Entities | Classes

Object property hierarchy: hasPart

Asserted

owl:topObjectProperty
hasPart

Annotations | Usage

Annotations: hasPart

Annotations

Characteri

Functional
Inverse functio
Transitive
Symmetric
Asymmetric
Reflexive
Irreflexive

Description: hasPart

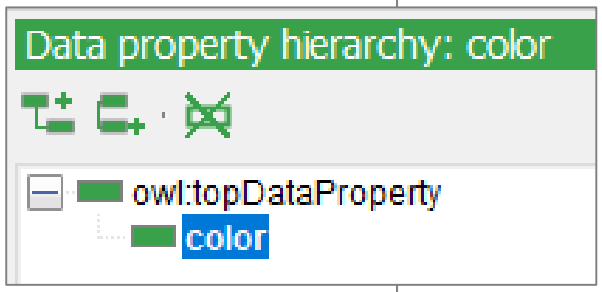
Equivalent To
SubProperty Of
Inverse Of
Domains (intersection)
Seat
Ranges (intersection)
Part
Disjoint With
SuperProperty Of (Chain)

To use the reasoner click Reasoner > Start reasoner Show Inferences

Building an OWL Ontology using Protégé: Data Properties

relationships between an individual and data values.

MyChair color: “brown”



untitled-ontology-22 (http://www.semanticweb.org/croche/ontologie...

File Edit View Reasoner Tools Refactor Window Help

untitled-ontology-22 Search...

color

Individuals by class | OntoGraf | SPARQL Query | Data properties

Active ontology | Entities | Classes | Object properties

Data property hierarchy: color

Asserted

owl:topDataProperty

color

Annotations | Usage

Annotations: color

Annotations

Characteristics

Functional

Description: color

Equivalent To

SubProperty Of

Domains (intersection)

Seat

Ranges

xsd:string

Disjoint With

To use the reasoner click Reasoner > Start reasoner

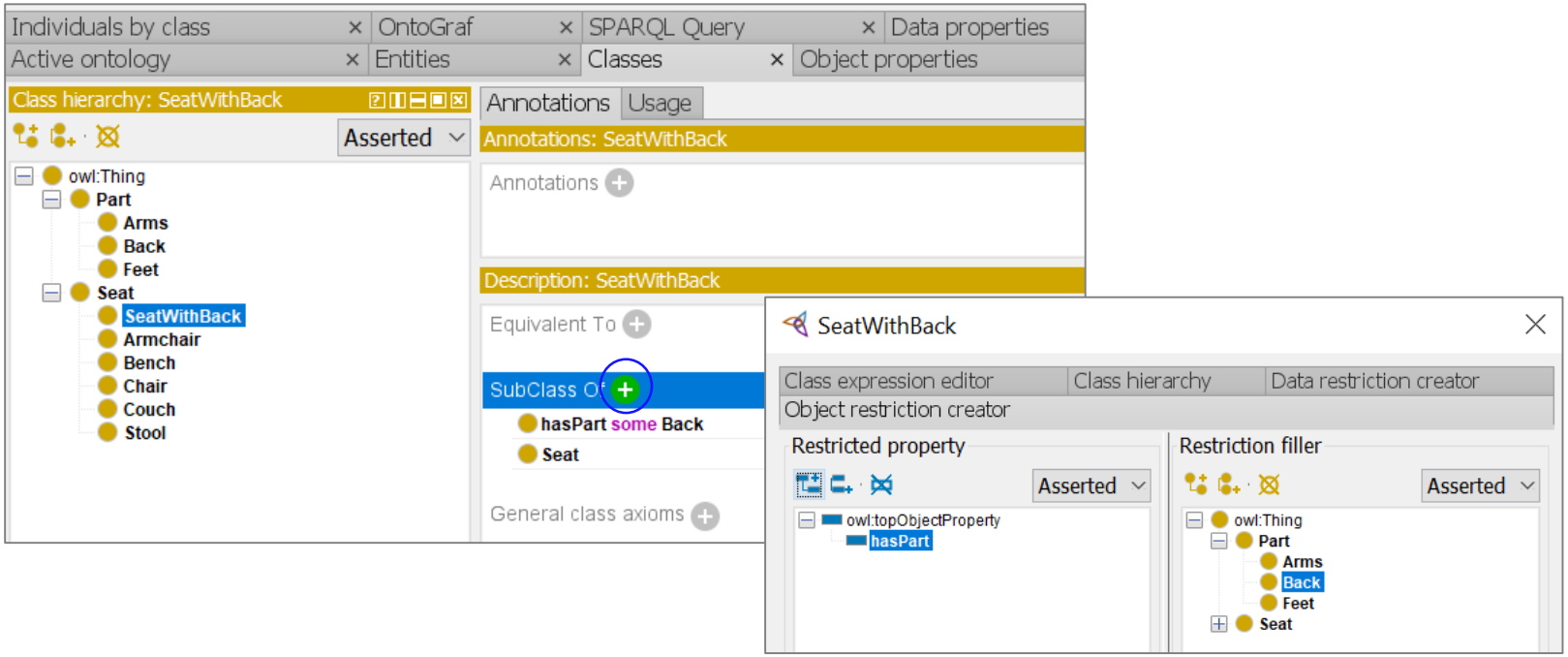
Show Inferences

Building an OWL Ontology using Protégé: Properties Restrictions

Existential Restrictions

An **existential restriction** describes a **class of individuals** that have **at least one (some) relationship** along a **specified property** to an **individual** that is a member of a **specified class**.

SeatWithBack = { x / ∃ y Back(y) ∧ hasPart(x,y) }



Building an OWL Ontology using Protégé: Class Expressions

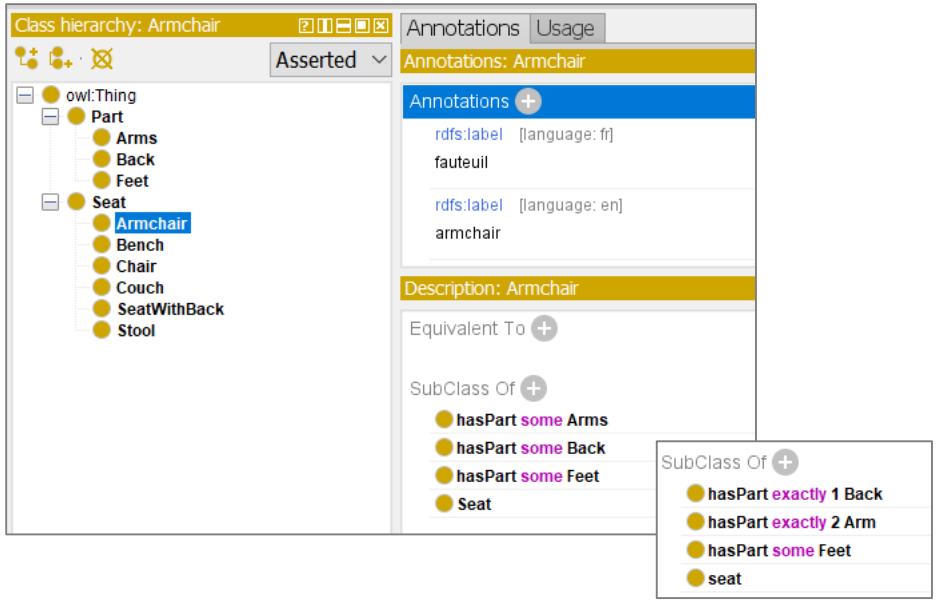
Class expressions are used to describe individuals that share common characteristics.

Class Expression Syntax

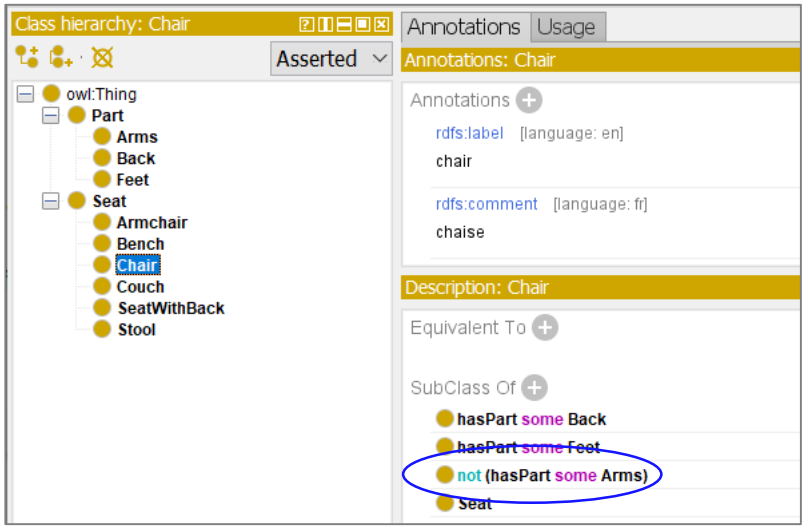
http://protegeproject.github.io/protege/class-expression-syntax/

Keyword	Example	Intuitive Meaning
some	hasPet some Dog	Things that have a pet that is a Dog This is the most common type of class expression. Also known as, an "SomeValueFrom restriction" or an "Existential Restriction". This kind of class expression consists of a property (in this case hasPet) and a class expression that is known as a filler (in this case the filler is Dog). Individuals that are instances of this class expression have a relationship along the hasPet property to an individual that is an instance of class Dog.
value	hasPet value Tibbs	Things that have a pet that is Tibbs. Here, Tibbs is a specific individual. Intuitively this would describe Tibb's owners. Note that this is a short cut for, and logically equivalent to, (hasPet some {Tibbs}), where the curly brackets describe a class of specific individuals - in this case, a class of one individual that is Tibbs. Also known as a "HasValue restriction"
only	hasPet only Cat	Things that have pets that are only Cats. Note that this does not mean that these things must have a Cat, but if they do have a pet then it will be a Cat. Also known as an "AllValuesFrom restriction" or a "Universal restriction"
min	hasPet min 3 Cat	Things that have at least three pets that are Cats. Things that have at least three pets that are Cats. Also known as a "Min cardinality restriction"

Essential Characteristics: Property Restriction

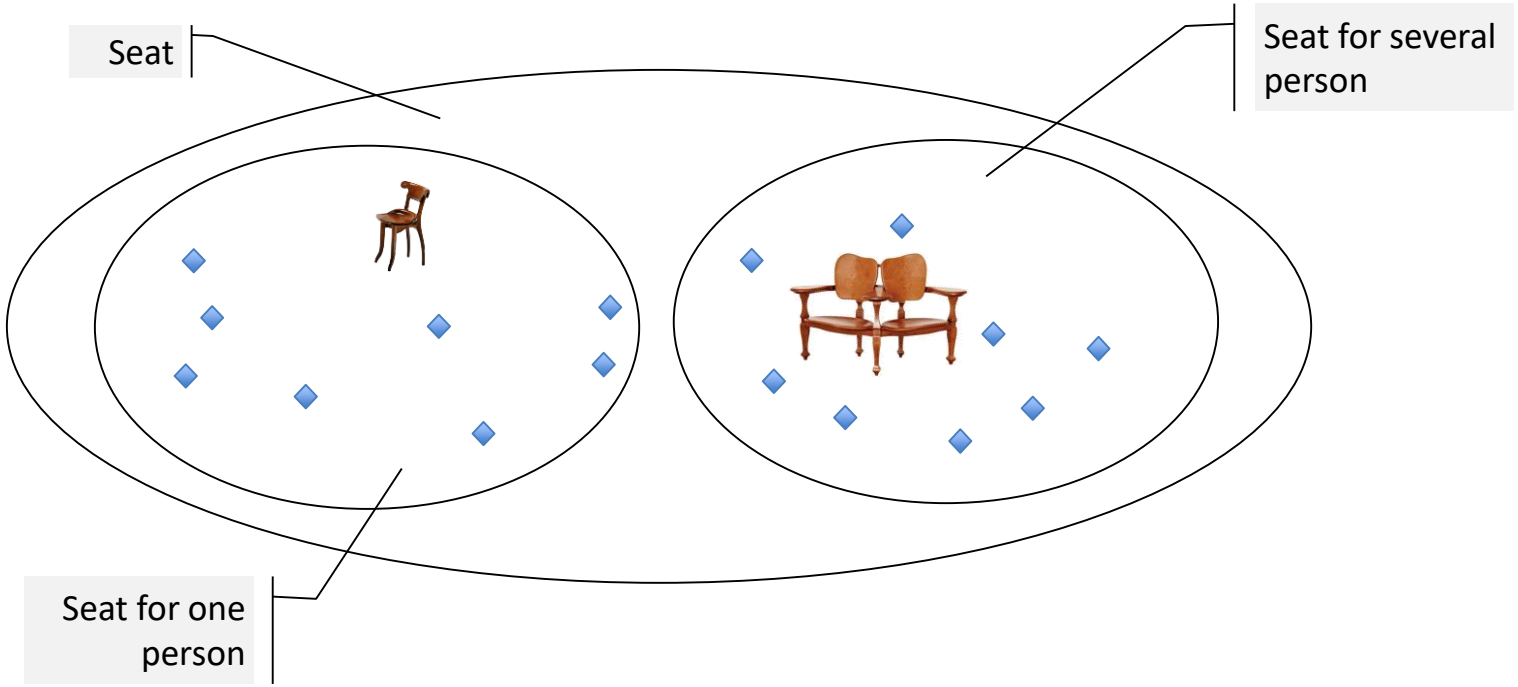


Without arms?



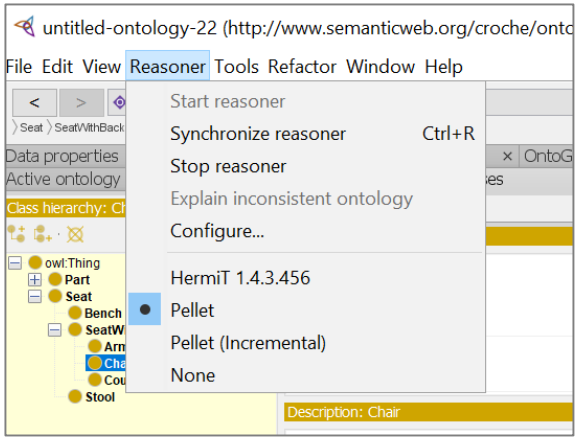
Essential Characteristics: Class

Seat for several persons?

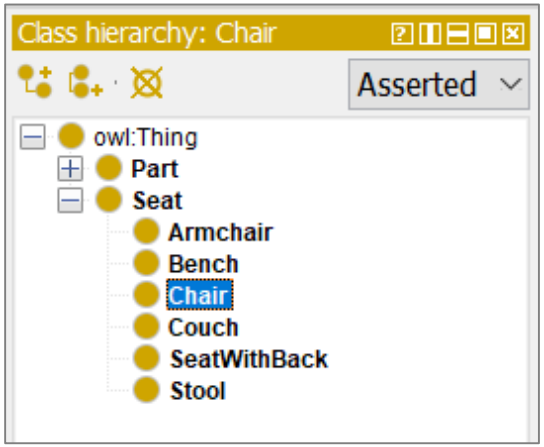


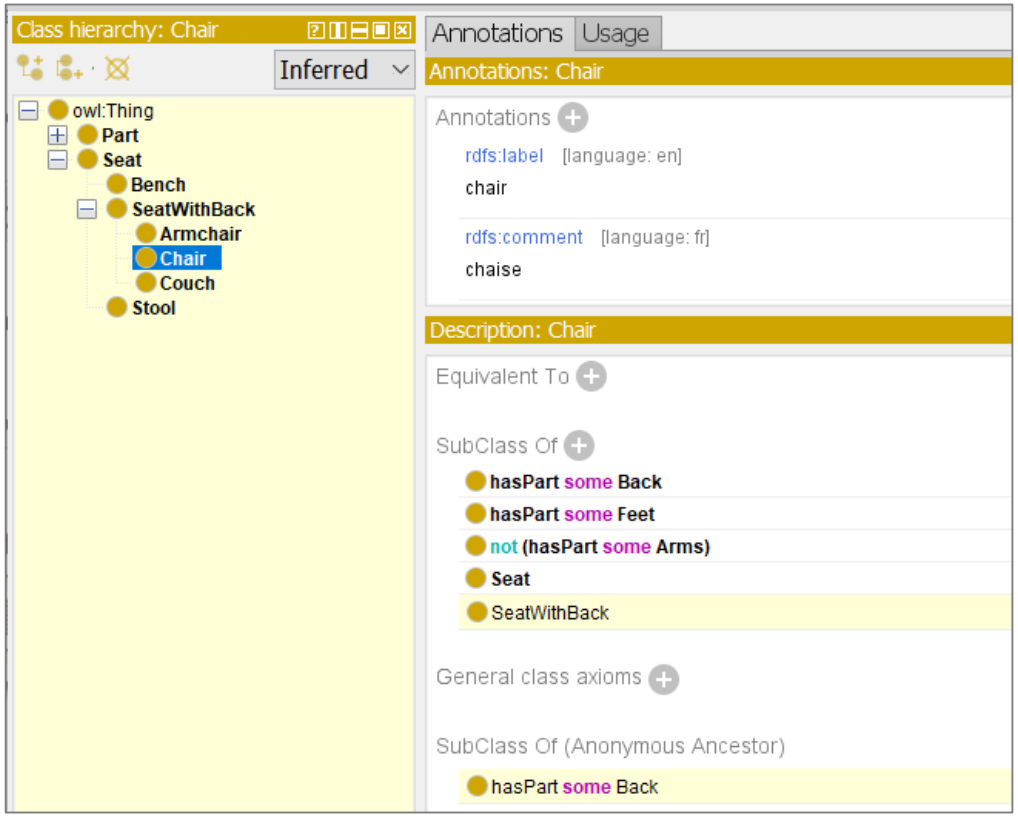
Building an OWL Ontology using Protégé: Reasoner

Protégé 4 allows different OWL reasoners to be plugged in

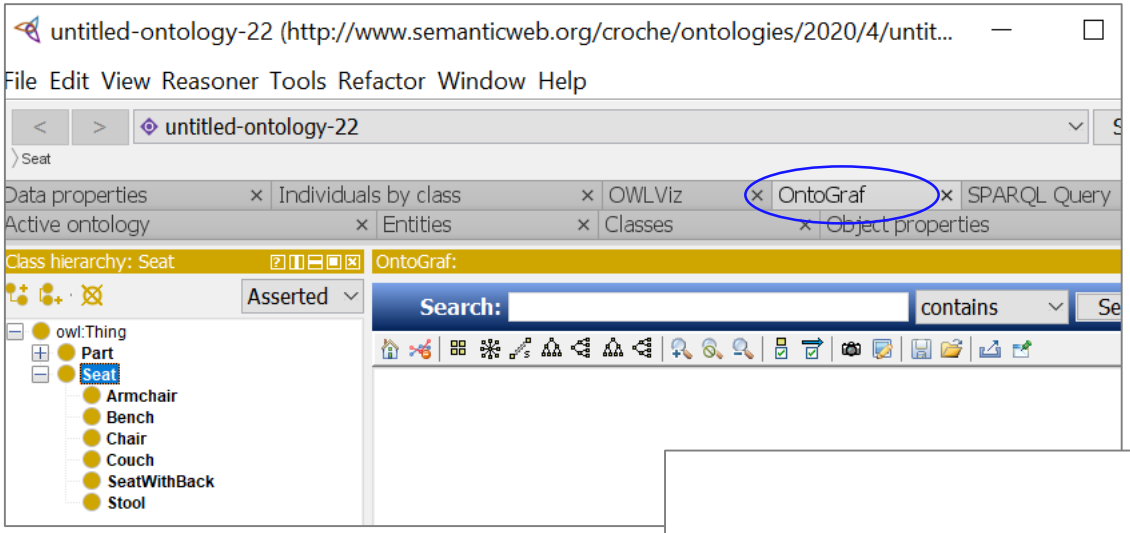


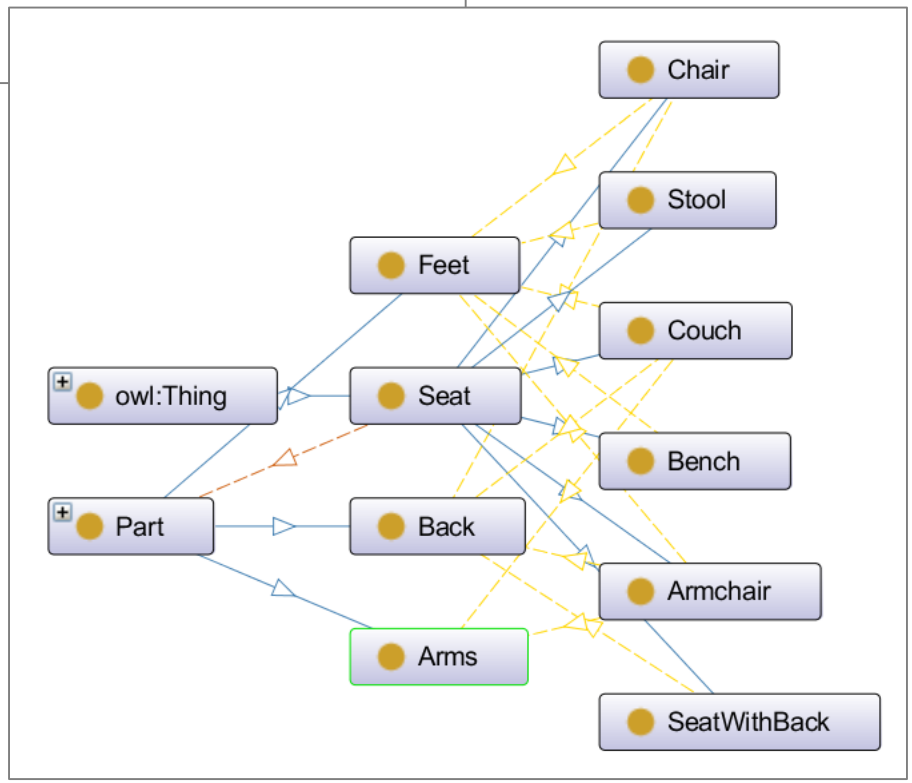
The class hierarchy that is automatically computed by the reasoner is called the inferred hierarchy.





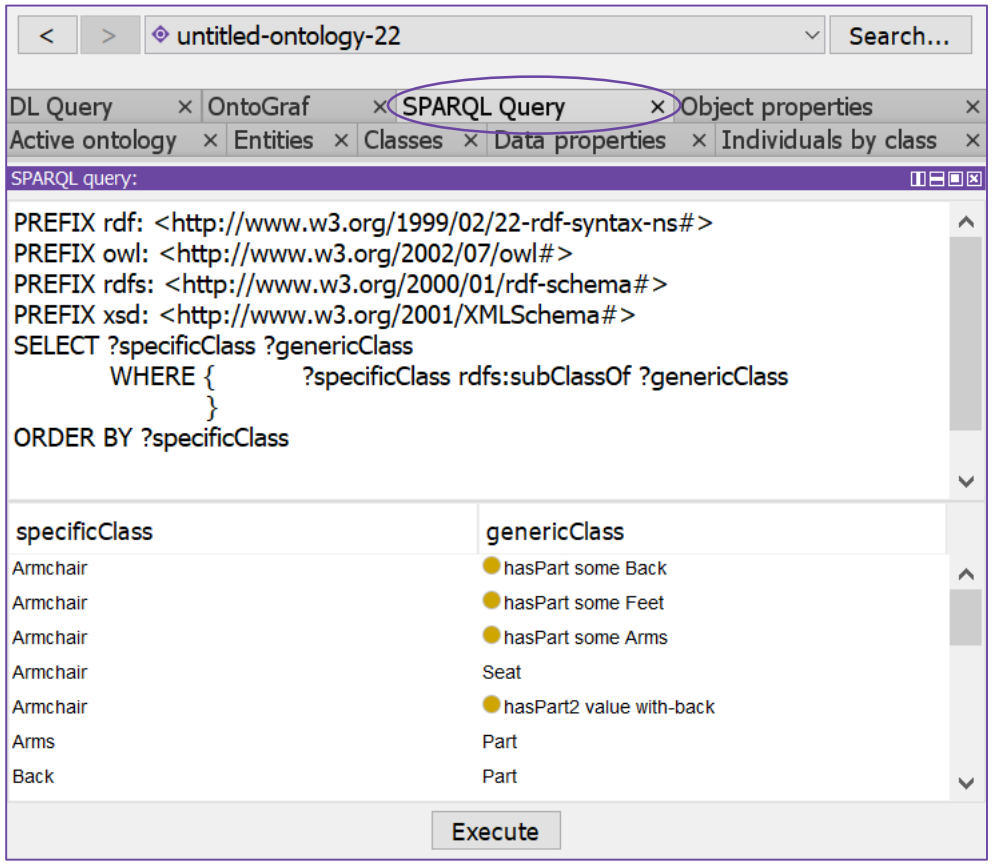
Building an OWL Ontology using Protégé: Visualisation



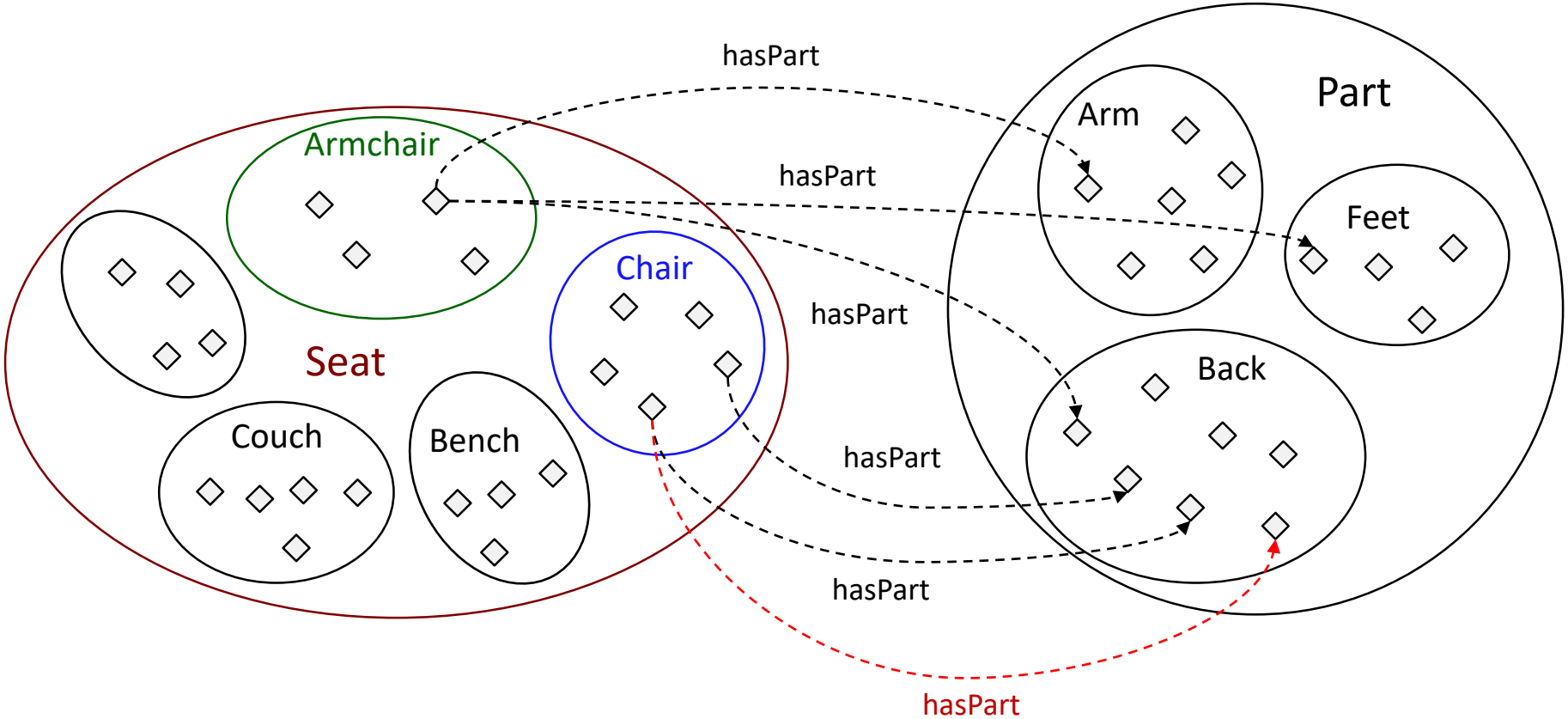


Building an OWL Ontology using Protégé: SPARQL Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?specificClass ?genericClass
        WHERE { ?specificClass rdfs:subClassOf ?genericClass }
ORDER BY ?specificClass
```



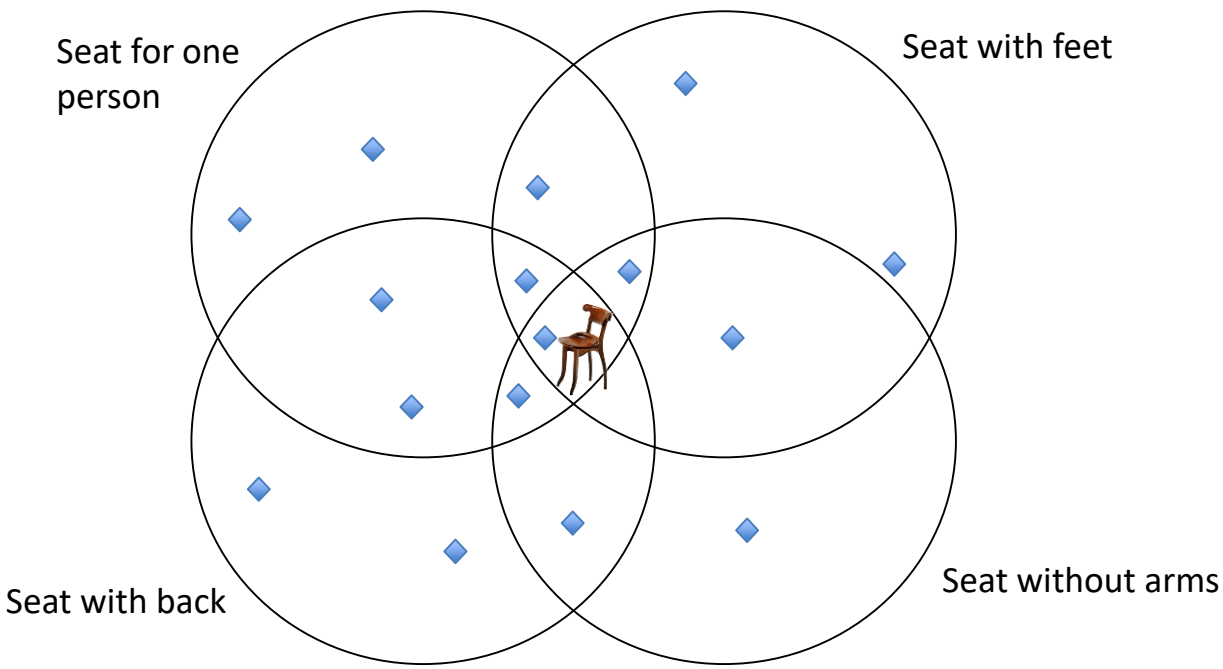
Building an OWL Ontology using Protégé: Essential Characteristics



Seat for one person?

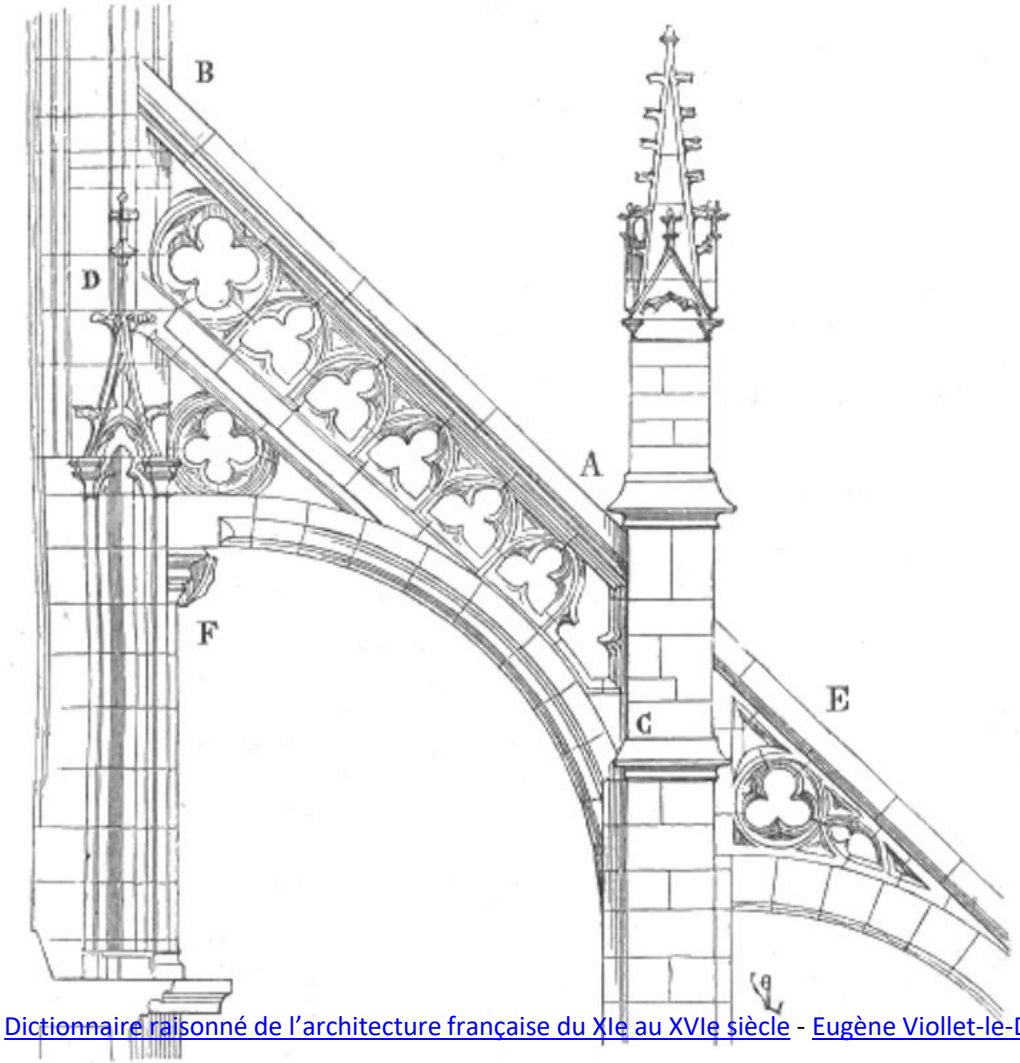
Building an OWL Ontology using Protégé: Essential Characteristics

- Essential characteristic == Class
- Essential characteristic == Role restriction
- Essential characteristic == individual



OntoTerminology Editor

http://ontoterminology.com/ May 2020



Dictionnaire raisonné de l'architecture française du XIe au XVIe siècle - Eugène Viollet-le-Duc

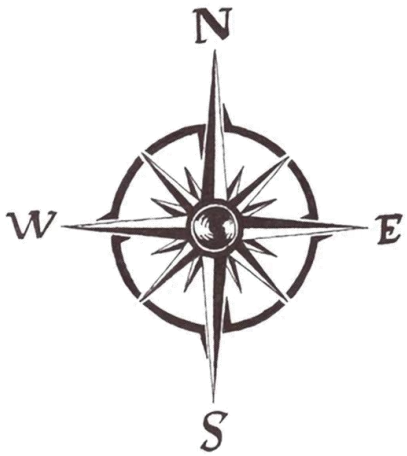
Prof. Christophe Roche

Condillac Research Group – LISTIC Lab.
University Savoie Mont-Blanc (France)

KETRC Research Centre
University of Liaocheng (China)

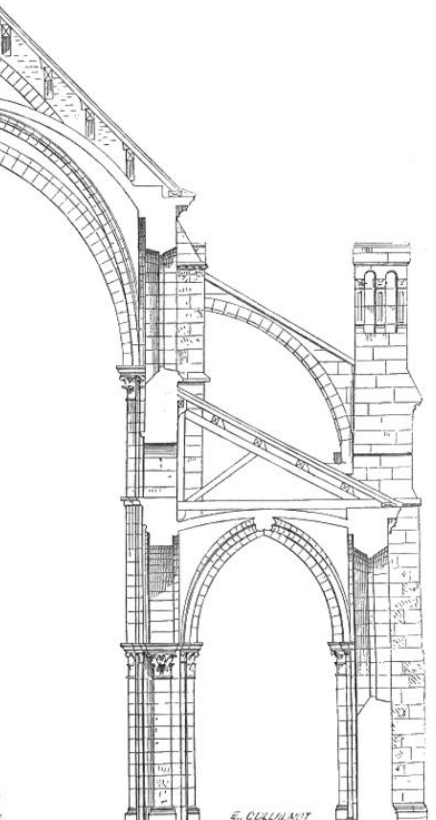
http://christophe-roche.fr/

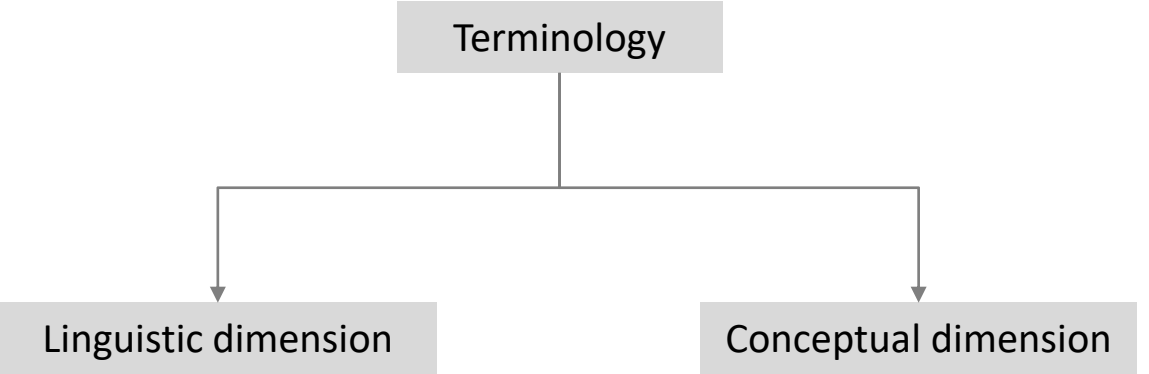
Contents



(1) Theoretical Foundations

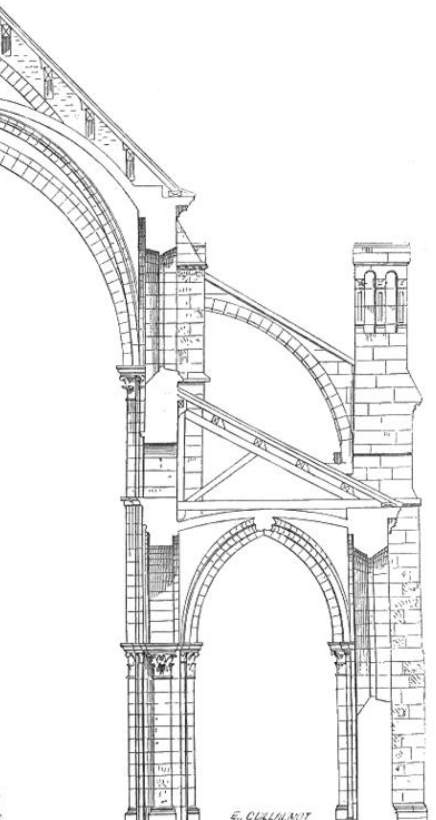
- An ontoterminology is a terminology whose conceptual system is a formal ontology

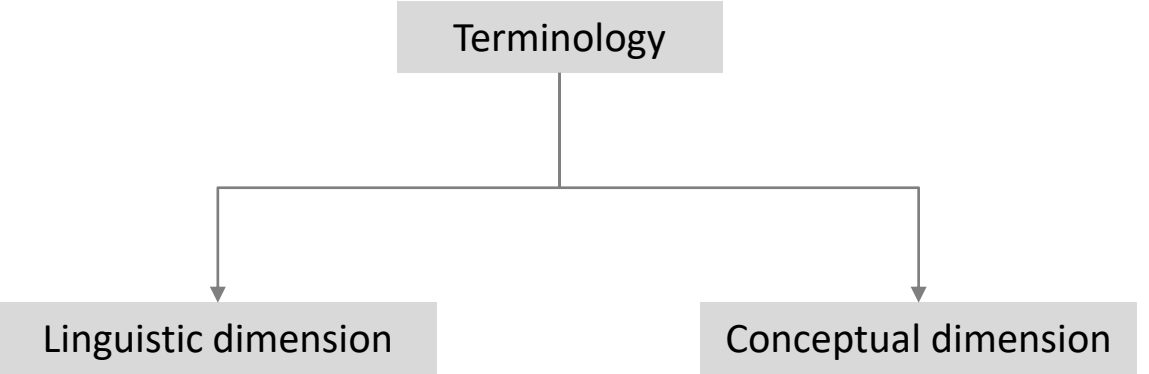




(1) Theoretical Foundations

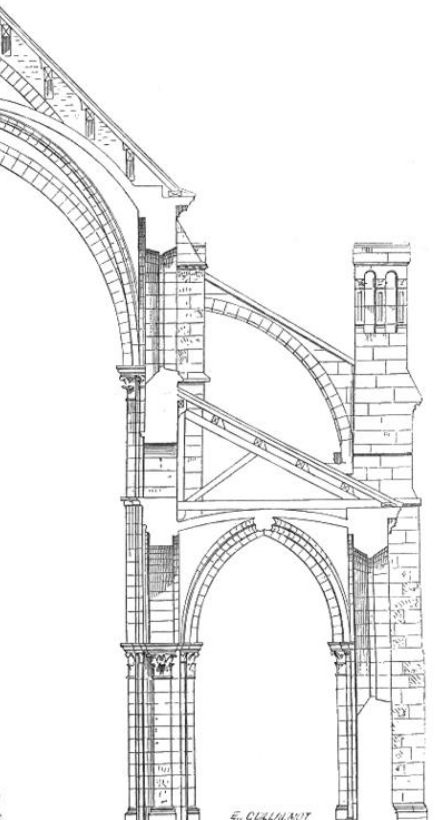
- An ontoterminology is a terminology whose conceptual system is a formal ontology
- A term is a verbal designation of a concept

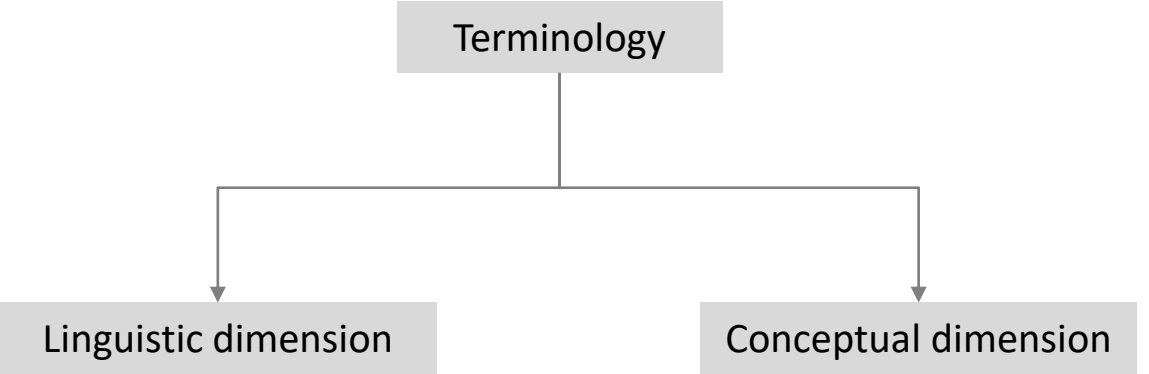




(1) Theoretical Foundations

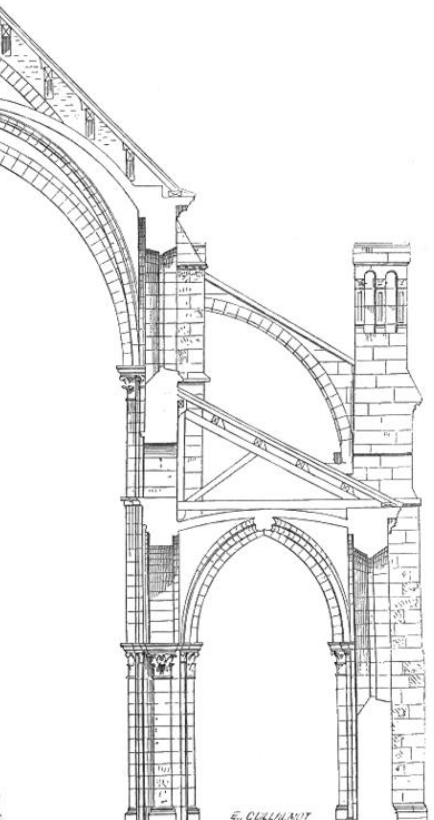
- An ontoterminology is a terminology whose conceptual system is a formal ontology
- A term is a verbal designation of a concept
- A concept is a unique combination of essential characteristics





(1) Theoretical Foundations

- An ontoterminology is a terminology whose conceptual system is a formal ontology
- A term is a verbal designation of a concept
- A concept is a unique combination of essential characteristics
- A concept is a set of essential characteristics enough stable to be named in a given natural language



Terminology

Linguistic dimension

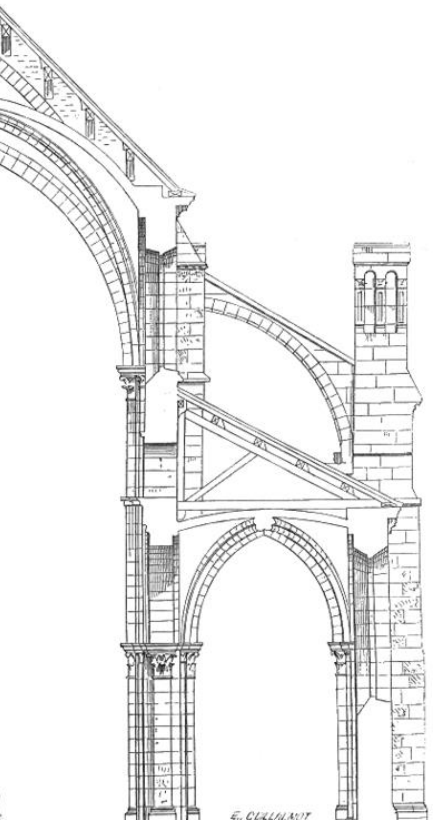
Conceptual dimension

(1) Theoretical Foundations

Conceptual dimension

- A concept is defined as a unique combination of essential characteristics

<Seat for one person> ::= <Seat> + /one person/
<Seat with feet> ::= <Seat> + /with feet/
<Seat with back> ::= <Seat> + /with back/
etc.

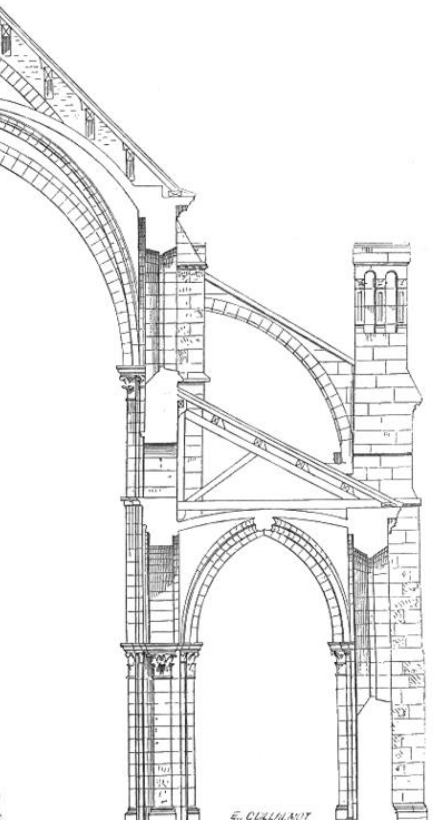


(1) Theoretical Foundations

Conceptual dimension

- A concept is defined as a unique combination of essential characteristics

<Seat for one person> ::= <Seat> + /one person/
<Seat with feet> ::= <Seat> + /with feet/
<Seat with back> ::= <Seat> + /with back/
etc.



- Essential characteristics are structured into axes of analysis and are exclusive each other

Back = { /with back/ /without back/ }
Feet = { /with feet/ /without feet/}
etc.

(1) Theoretical Foundations

Conceptual dimension

- A concept is defined as a unique combination of essential characteristics

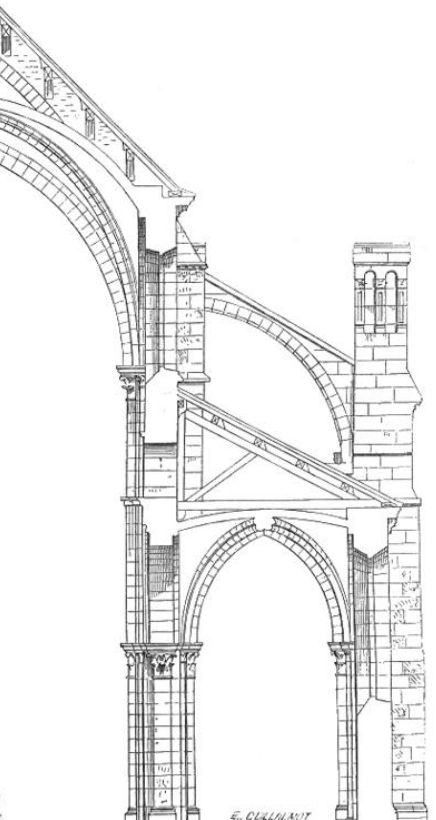
<Seat for one person> ::= <Seat> + /one person/
<Seat with feet> ::= <Seat> + /with feet/
<Seat with back> ::= <Seat> + /with back/
etc.

- Essential characteristics are structured into axes of analysis and are exclusive each other

Back = { /with back/ /without back/ }
Feet = { /with feet/ /without feet/}
etc.

- Multiple hierarchy => multiple inheritance

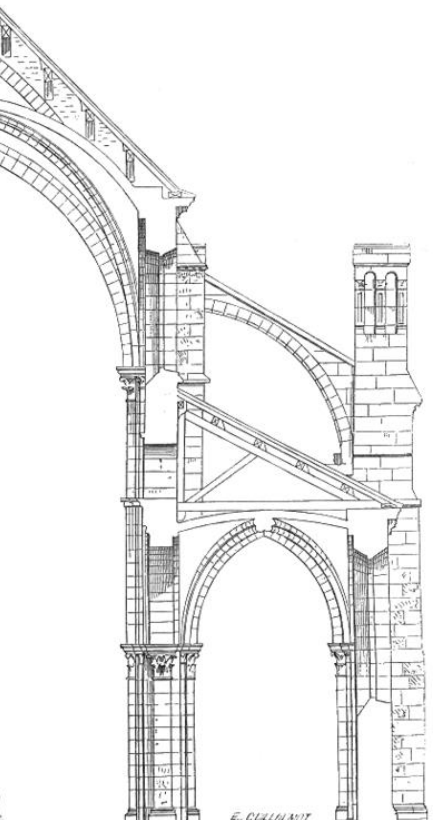
<Seat for one person with feet with back without arm>
is-a <Seat for one person>
is-a <Seat with feet>
is-a <Seat with back>
etc.

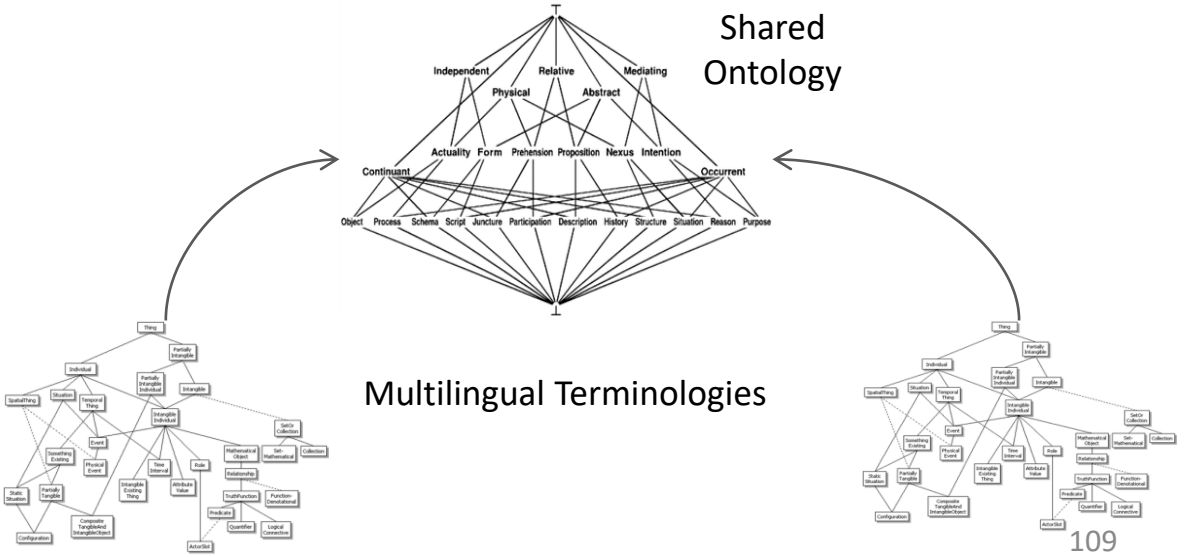


(1) Theoretical Foundations

Linguistic dimension

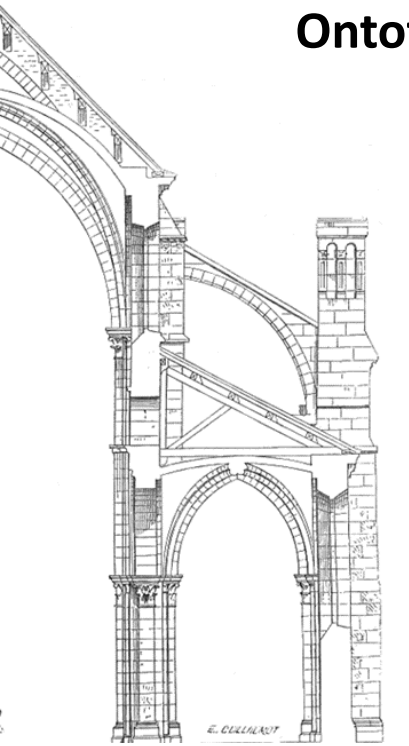
- Multi-lingual terminology
- Linguistic diversity
- Ontology shared by different terminologies
- Aristotelian definition in natural language (genus and specific difference) based on the formal definition

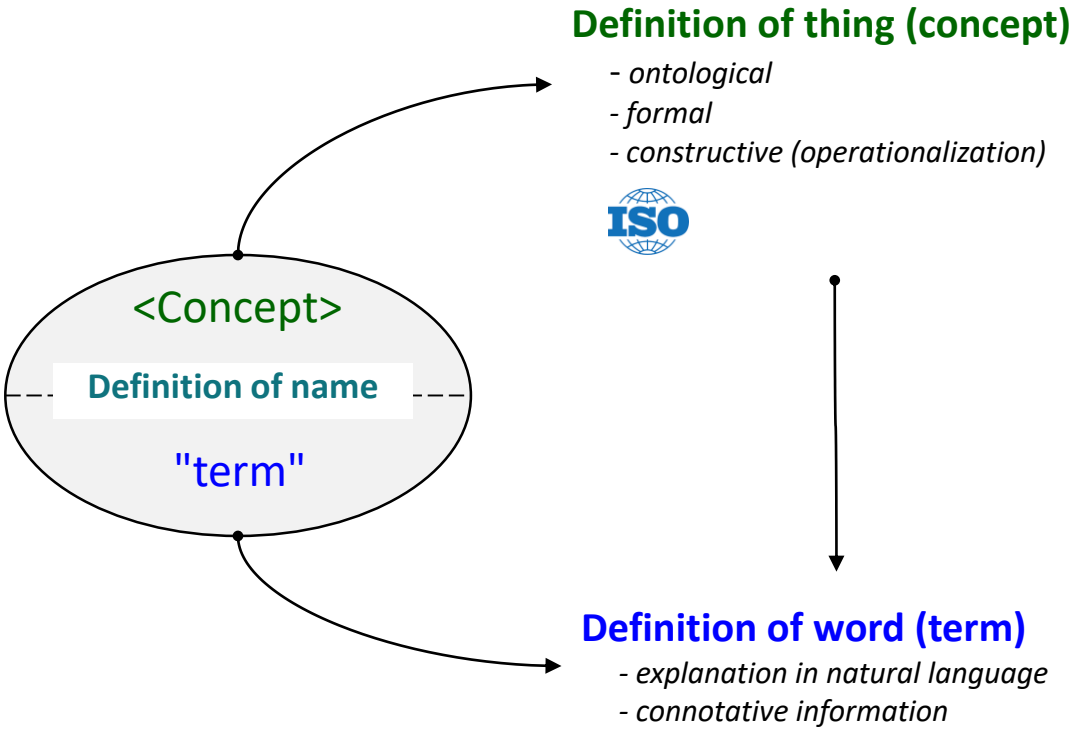




(1) Theoretical Foundations

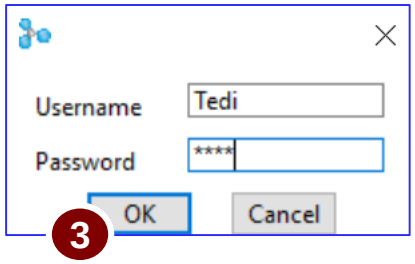
Ontoterm





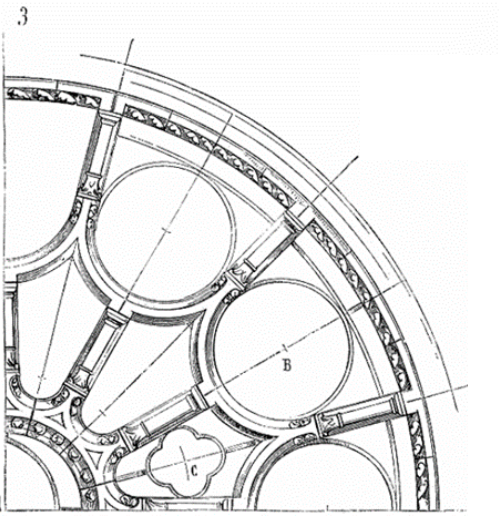
(2) Environment

- Opening a session



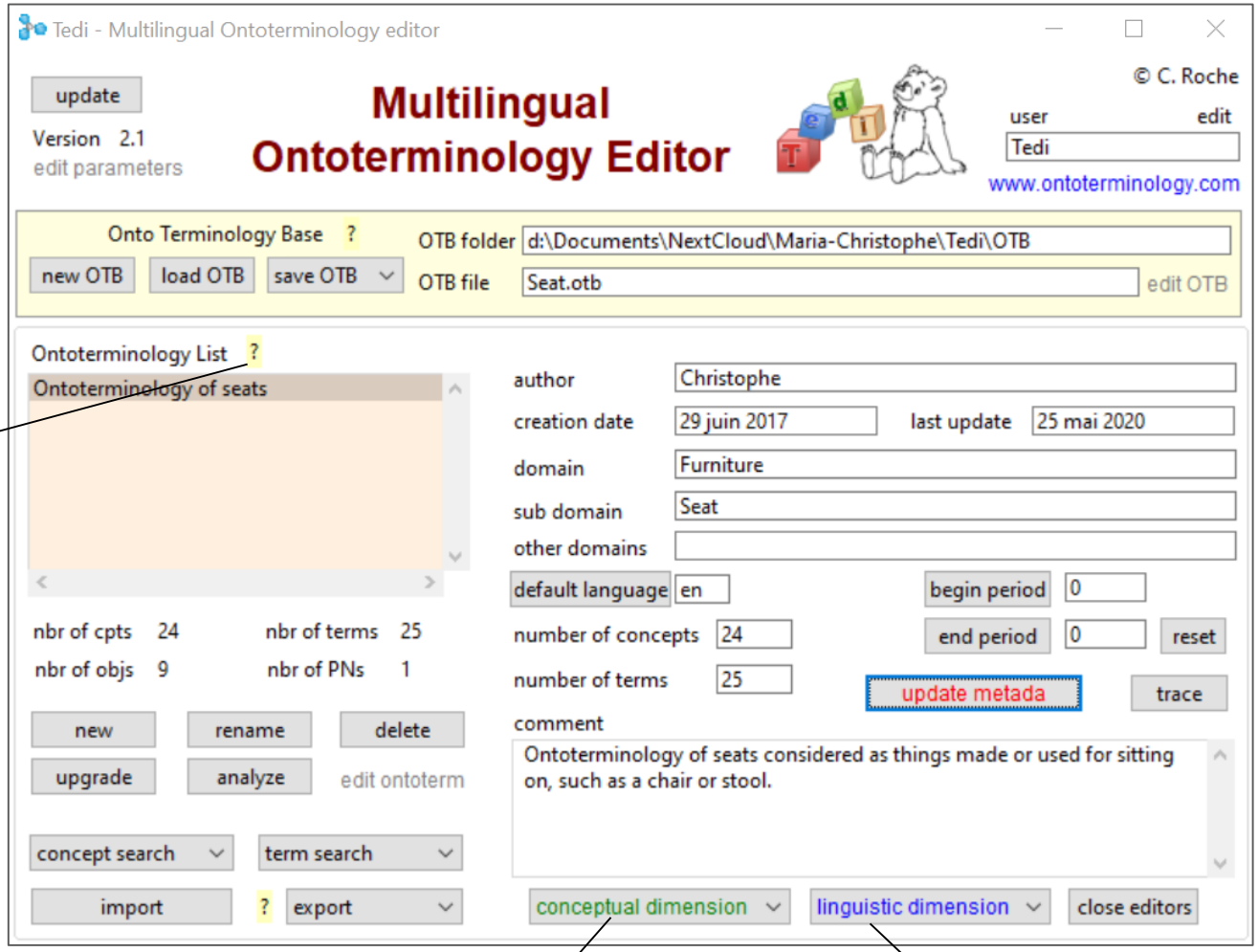
Tedi

Tedi



(2) Environment

- Launcher



Help functions

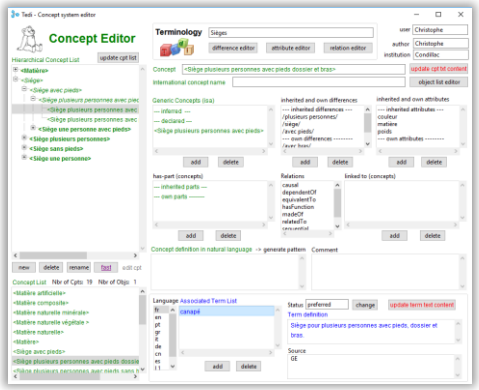
Access to the editors of the conceptual dimension

Access to the editors of the linguistic dimension

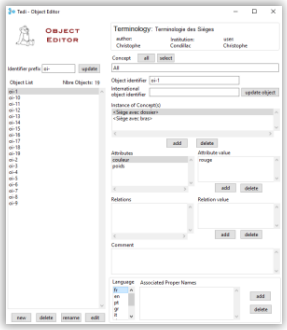
(2) Environment

- A set of dedicated editors accessible from the Tedi Launcher

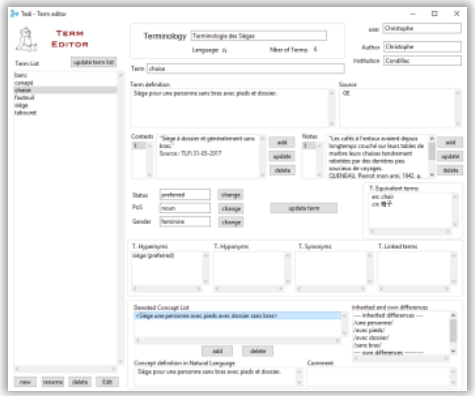
Concept editor



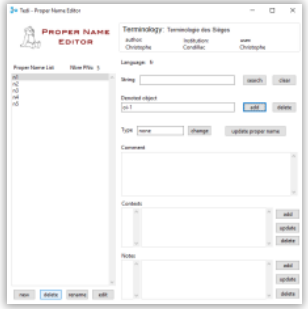
Object editor



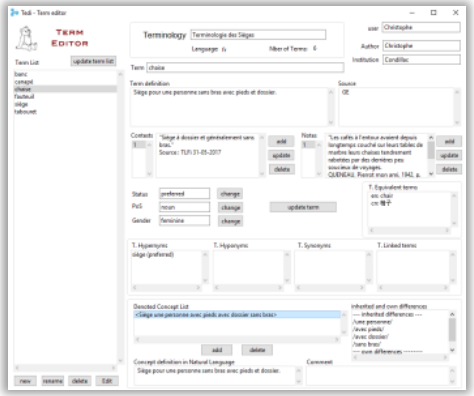
Term editor



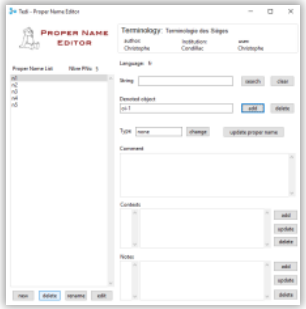
Proper names editor

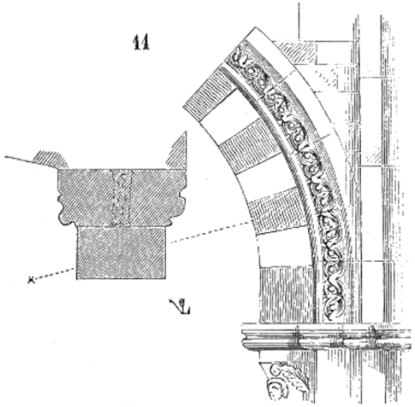


Term Editor

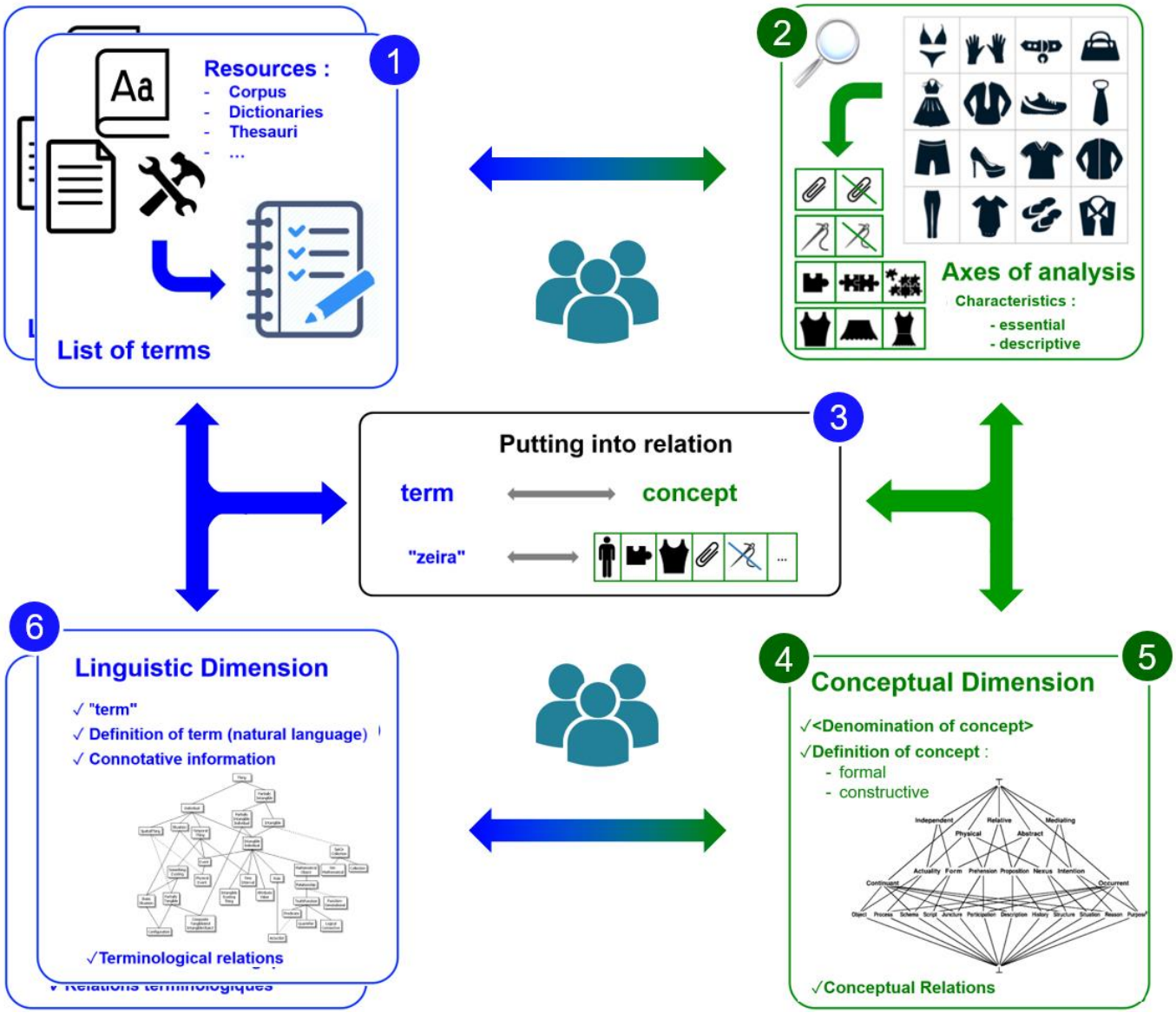


Proper names editor





(3) A Term-Guided Methodology

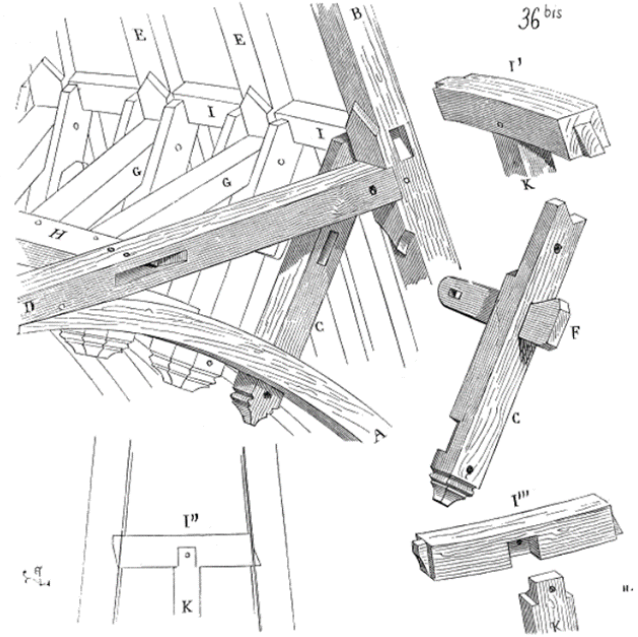


(3) A Term-Guided Methodology

- Experts are guided by Tedi which proposes only valid information at each step

1: Enter the terms to be defined

2: Identify the axes of analysis and their essential characteristics



3: Select the set of characteristics denoted by the term

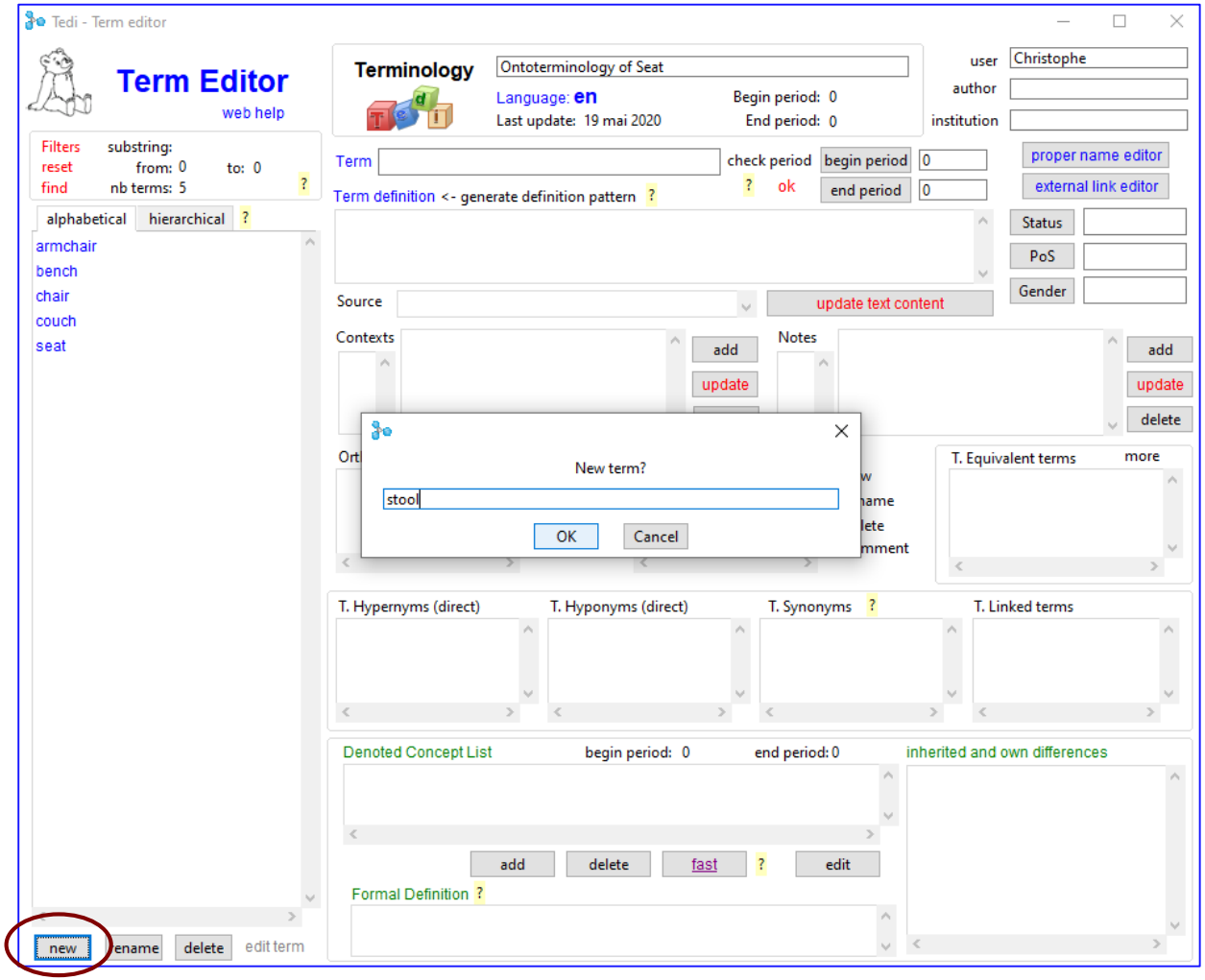
4: Create the concept if it does not exist

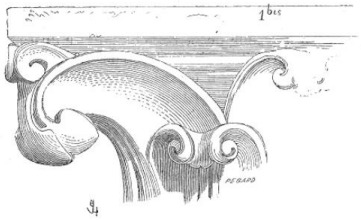
5: Update the concept system if necessary

6: Complete the linguistic dimension

(3) A Term-Guided Methodology

1 Enter the terms to be defined





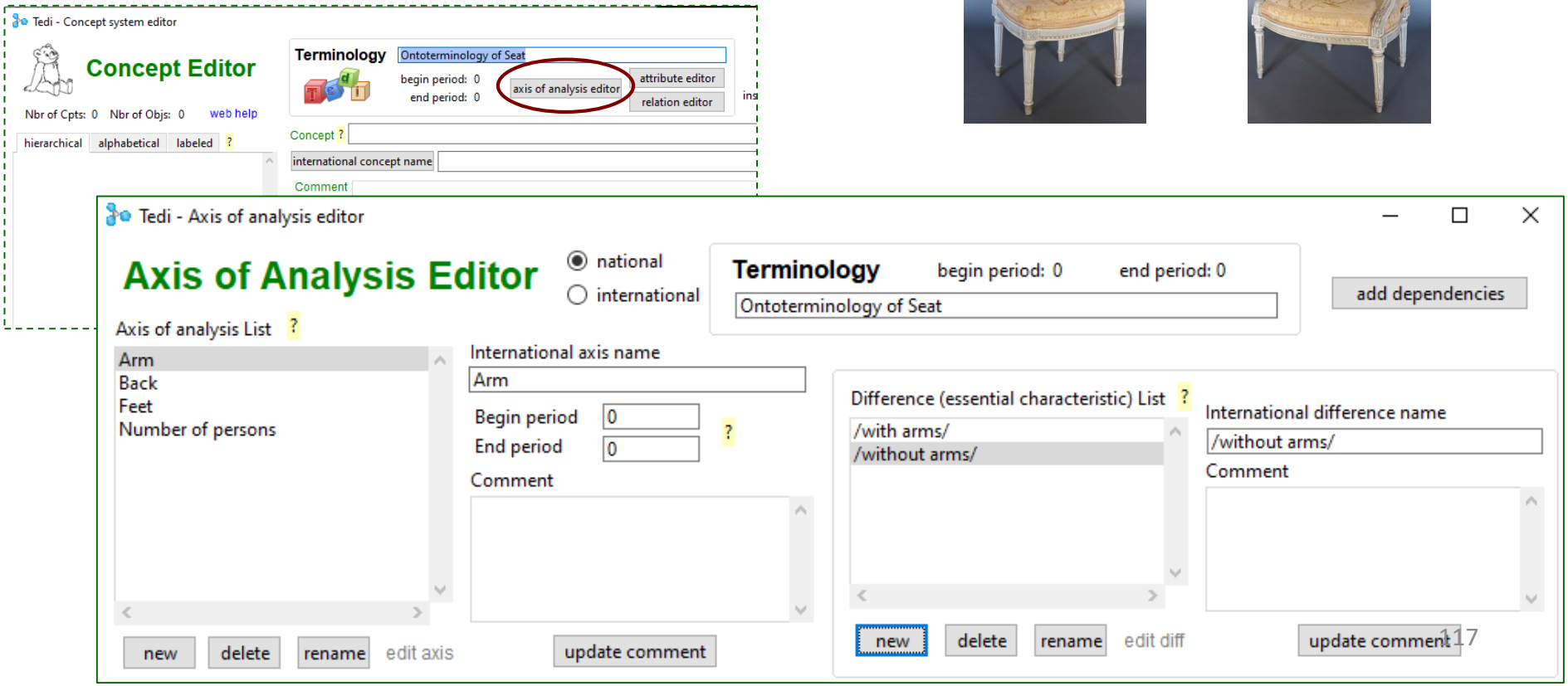
(3) A Term-Guided Methodology

2 Identify the Axes of Analysis and their essential characteristics

☞ Looking for differences between objects

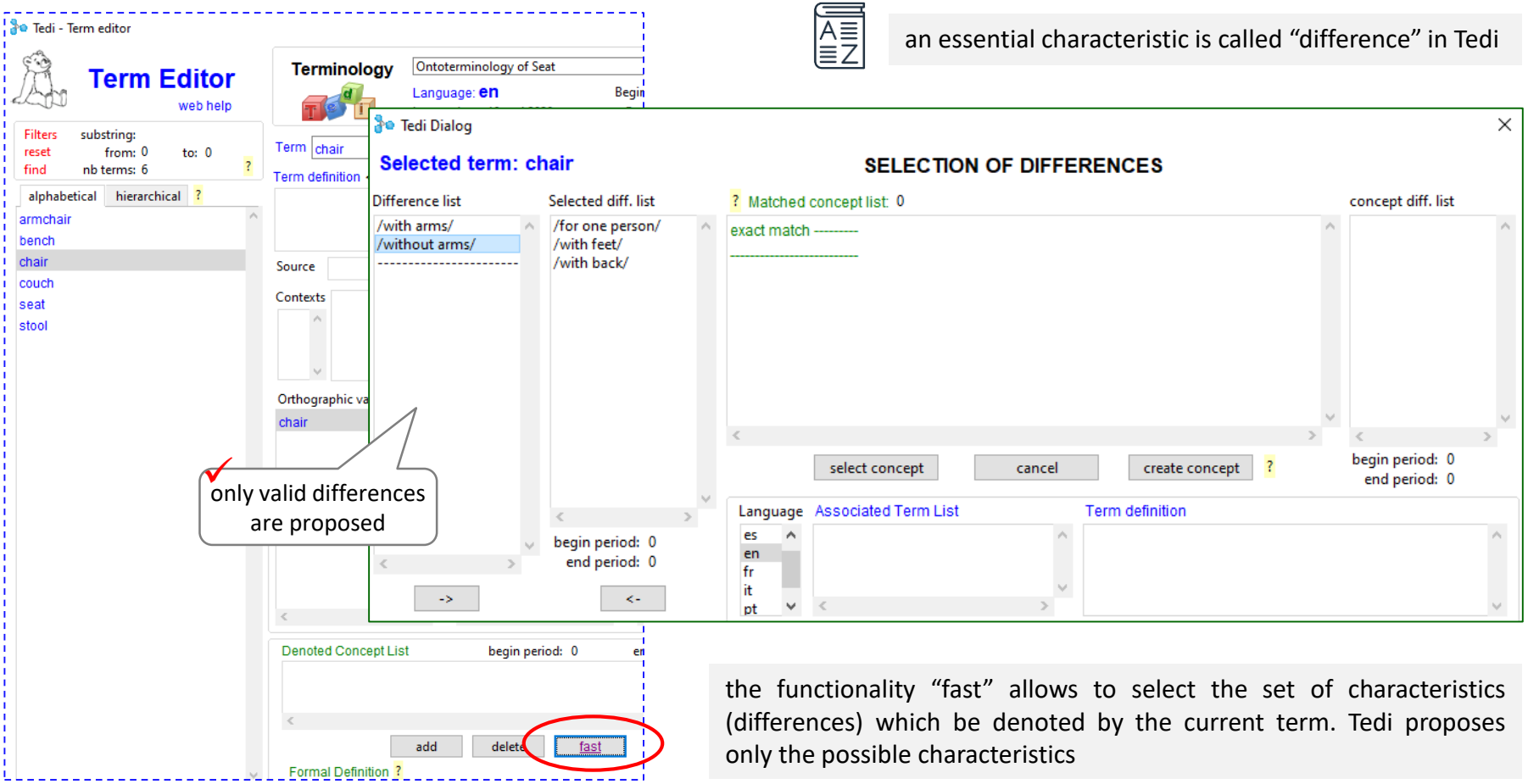
/without arms/

/with arms/



(3) A Term-Guided Methodology

3 Select the set of characteristics denoted by a term

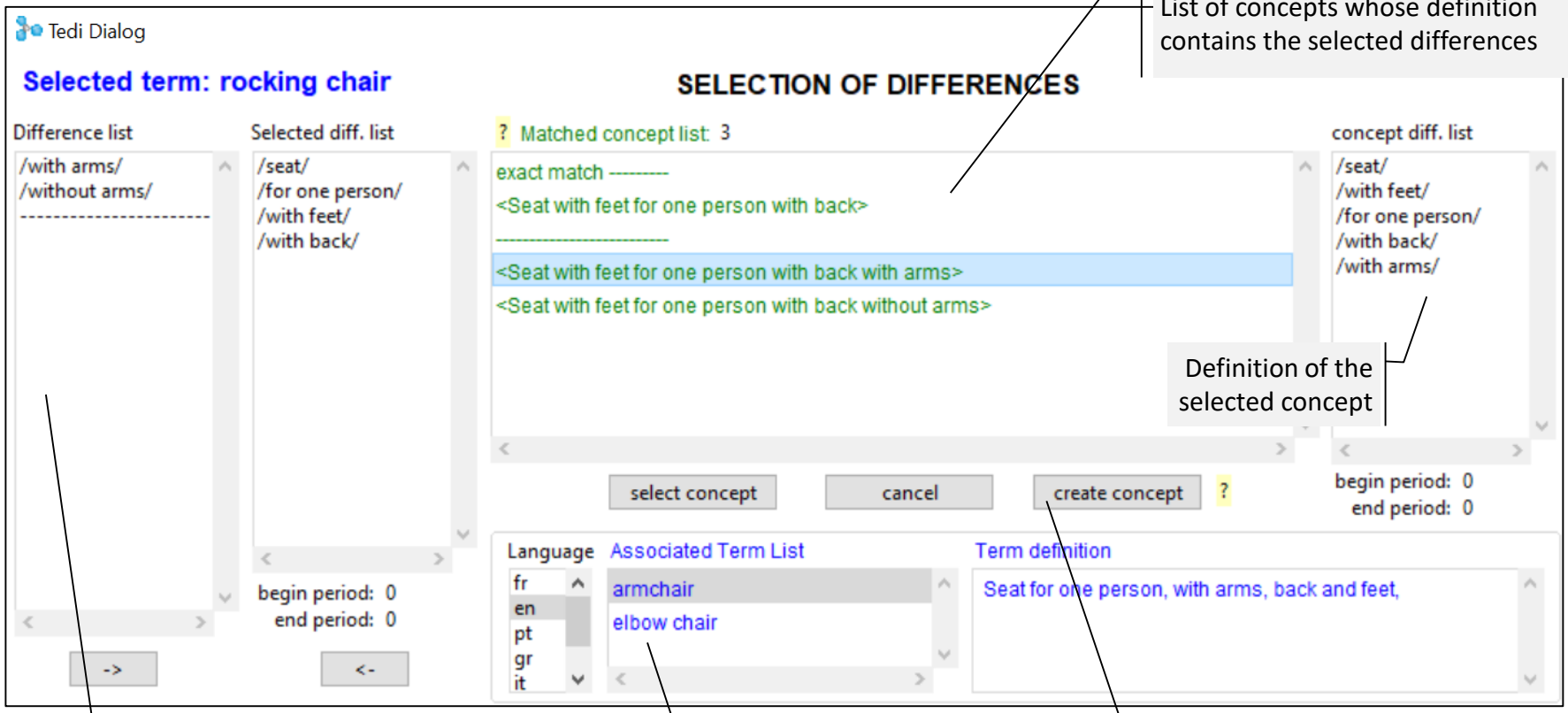


an essential characteristic is called “difference” in Tedi

the functionality “fast” allows to select the set of characteristics (differences) which be denoted by the current term. Tedi proposes only the possible characteristics

(3) A Term-Guided Methodology

- Connecting Terms and Concepts
  - Terms are known to experts
  - A term denotes a set of essential characteristics stable enough to have a name in a given natural language



List of concepts whose definition contains the selected differences

Definition of the selected concept

List of possible differences to choose from organized into axes of analysis

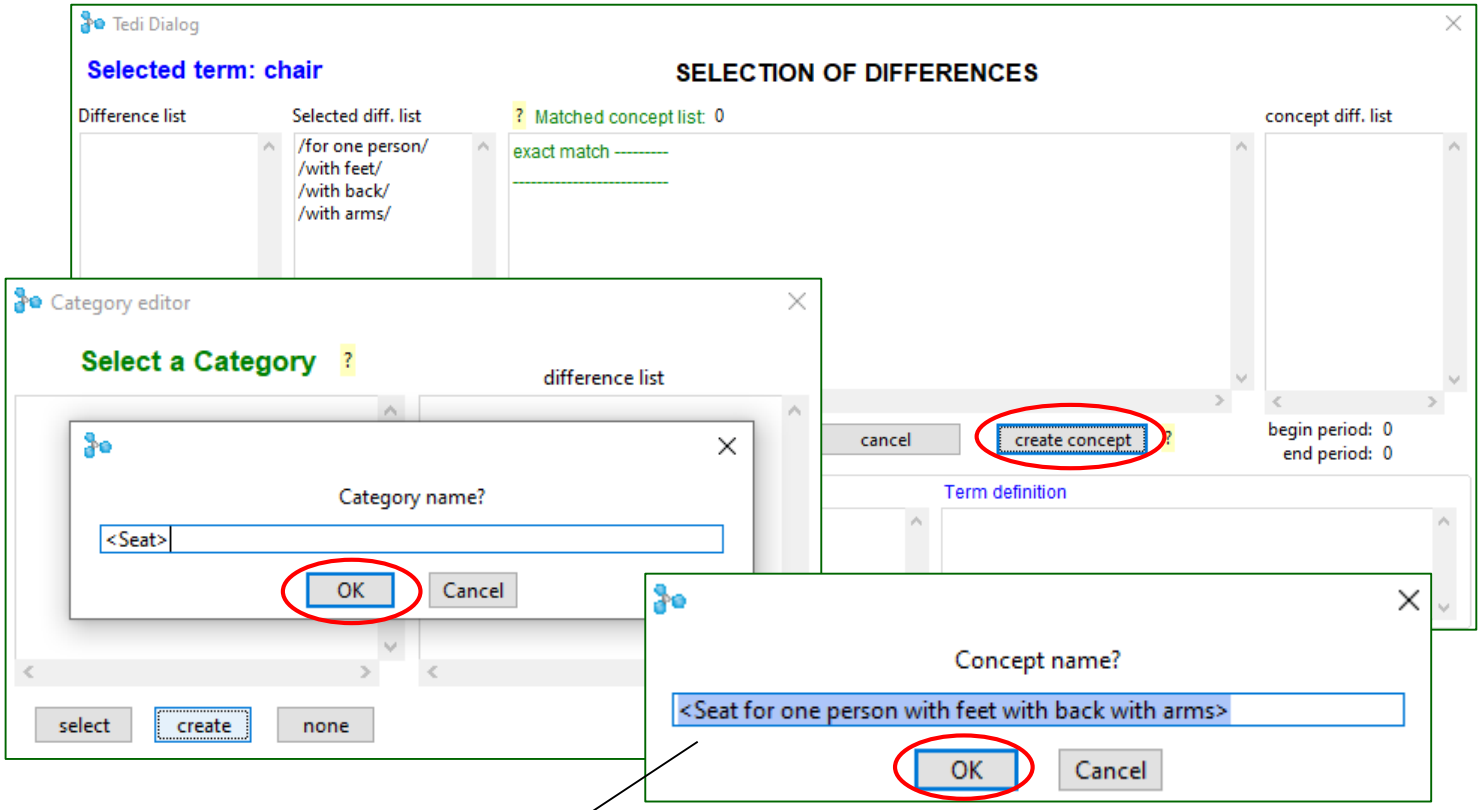
List of terms that denote the selected concept

Creation of new concept, if no exact match

(3) A Term-Guided Methodology

4 Create the concept if it does not exist

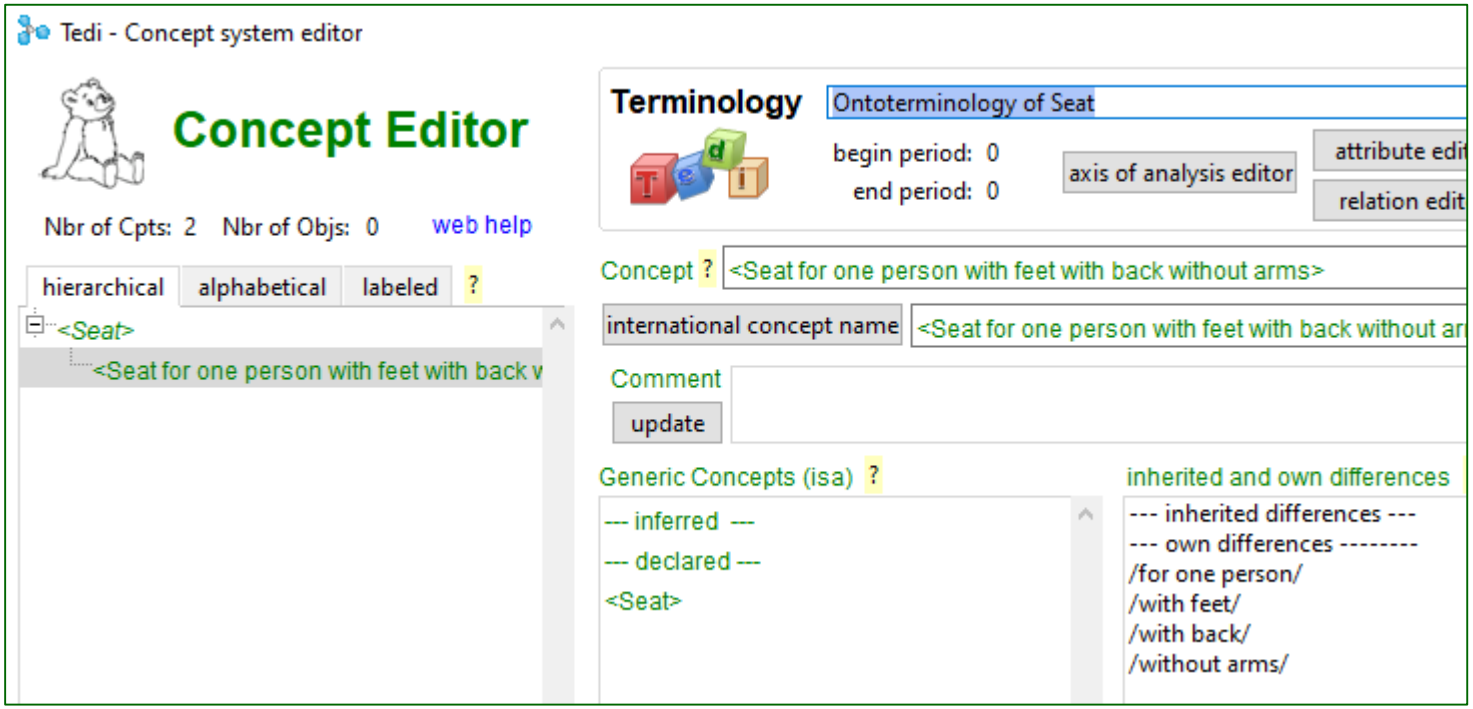
Each concept belongs to a category. Create it if it does not exist.



Automatic creation of concept name

(3) A Term-Guided Methodology

5 Update the concept system if necessary



Dedicated Editors

Tedi proposes only the valid options (checking « on the fly »)

Tedi - Concept system editor

Concept Editor

Nbr of Cpts: 24 Nbr of Objs: 9 web help

hierarchical alphabetical labeled ?

- <Material>
- <Part>
- <Seat>
  - <Seat for one person>
    - <Seat with feet for one person>
      - <Seat with feet for one person with
        - <Seat with feet for one person v
        - <Seat with feet for one person v
        - <Seat with feet for one person with
        - <Seat without feet for one person with
- <Seat for several persons>
- <Seat with feet>
- <Seat without feet>

new delete rename fast edit cpt

Terminology Ontoterminology of seats

begin period: 0
end period: 0

axis of analysis editor | attribute editor | relation editor

user Tedi
author Christophe
institution Condillac

Concept ? <Seat with feet for one person with back without arms>

links-illustration

international concept name <Seat with feets for one person with back without arms>

object list editor

Comment

update

begin period: 0
? end period: 0

Generic Concepts (isa) ?

--- inferred ---
--- declared ---
<Seat with feet for one person with back>

add delete

inherited and own differences ?

--- inherited differences ---
/seat/
/with feet/
/for one person/
/with back/
--- own differences --------
/without arms/

add delete

inherited and own attributes

--- inherited attributes ---
weight
material
colour
--- own attributes --------

add delete

has-part (concepts)

--- inherited parts ---
<Feet>
--- own parts --------
<Back>

add delete

Relations

causal
dependentOf
equivalentTo
hasFunction
madeOf

linked to (concepts)

add delete

Default list of relations

Formal Definition ? <Seat with feet for one person with back> + /without arms/

Language Associated Term List

fr
en
pt
gr
it
de
cn

chair

add delete edit

Status preferred change update

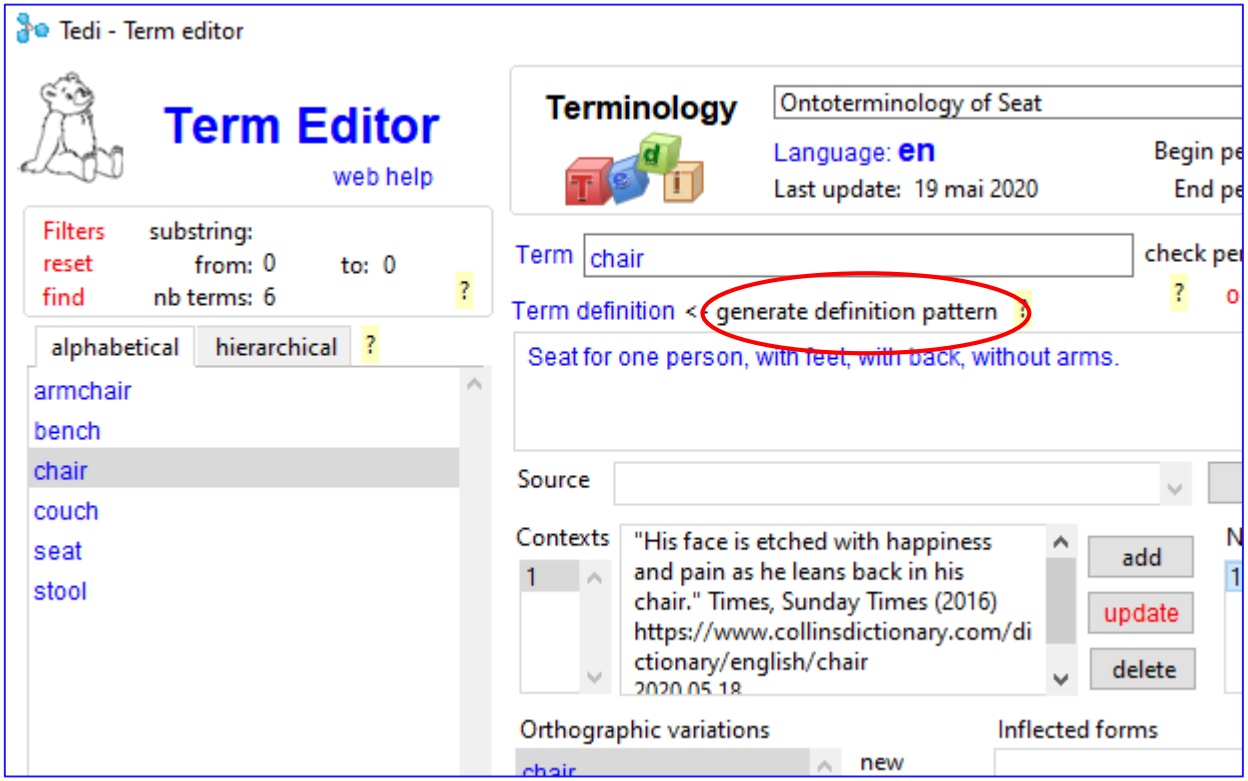
Term definition

Seat for one person, with back and legs, without arms.

Source

(3) A Term-Guided Methodology

6 Complete the linguistic dimension as necessary



Automatic generation of patterns of definition in natural language

Tedi - Term editor

Term Editor

web help

Filters substring:
reset from: 0 to: 0
find nb terms: 8 ?

alphabetical hierarchical ?

armchair
bench
chair
couch
elbow chair
ottoman
seat
stool

new rename delete edit term

Terminology

Ontoterminology of seats

Language: en
Last update: 25 mai 2020
Begin period: 0
End period: 0

user Tedi
author Christophe
institution Condillac

Term chair
check period begin period 0
? ok end period 0

proper name editor
external link editor

Term definition <- generate definition pattern ?

Seat for one person, with back and legs, without arms.

Status preferred
PoS noun
Gender neuter

Source
update text content

Contexts
1
'Cafe tables and chairs face a bench seat below a lowered ceiling that curves for an atmospheric effect.' https://www.lexico.com/en/definition/chair
2020.05.25
add update delete

Notes
1
"A chair is a piece of furniture for one person to sit on. Chairs have a back and four legs." https://www.collinsdictionary.com/dictionary/english/chair
2020.05.17
add update delete

Orthographic variations
chair
new rename delete comment

Inflected forms
chairs
new rename delete comment

T. Equivalent terms more
fr: chaise
fr: chaise d'intérieur
gr: καρέκλα

T. Hypernyms (direct)
seat (preferred)

T. Hyponyms (direct)

T. Synonyms ?

T. Linked terms

Automatic calculation of terminological equivalents in other languages (multilingualism)

Automatic calculation of terminological hypernyms, hyponyms, synonyms for every term

Denoted Concept List begin peri
<Seat with feet for one person with back without arms>
add delete fast ? edit

Formal Definition ?
<Seat with feet for one person with back> + /without arms/

inherited and own differences
--- inherited differences ---
/seat/
/with feet/
/for one person/
/with back/
--- own differences --------
/without arms/

(3) A Term-Guided Methodology

Generation of definition pattern in natural language

editing by the expert

term

"chair"

:

definition of the term in natural language

"Seat for one person with feet ~~with~~ and back without arms.

designates

"seat"

pattern generation

Linguistic Dimension

Conceptual Dimension

is designated by

concept

<Seat for one person with feet with back without arms>

definition of the concept in a formal language

::=

```
<Seat>
    + /for one person/
    + /with feet/
    + /with back/
    + /without arms/
```

(4) Export: HTML

Tedi Onto-Dictionary on "Ontoterminology of seats" (en)

Date: 25 mai 2020 - Time: 13:01:27 - Version: 2.1 - www.ontoterminology.com/tedi

search:

- armchair
- bench
- chair
- couch
- elbow chair
- ottoman
- seat
- stool

chair

Definition: Seat for one person, with back and legs, without arms.
Status: preferred
Context(s):
1) 'Cafe tables and chairs face a bench seat below a lowered ceiling that curves for an atmospheric effect.'
https://www.lexico.com/en/definition/chair
2020 05 25
Note(s):
1) "A chair is a piece of furniture for one person to sit on. Chairs have a back and four legs."
https://www.collinsdictionary.com/dictionary/english/chair
2020 05 17

Equivalent(s):
- fr: chaise (preferred)
- fr: chaise d'intérieur (alternative)
- gr: καρέκλα (preferred)

Concept: <Seat with feet for one person with back without arms>
essential characteristic(s): /seat/, /with feet/, /for one person/, /with back/, /without arms/,
a kind of: <Seat with feet for one person with back>,
linked to: <Back>, <Feet>,

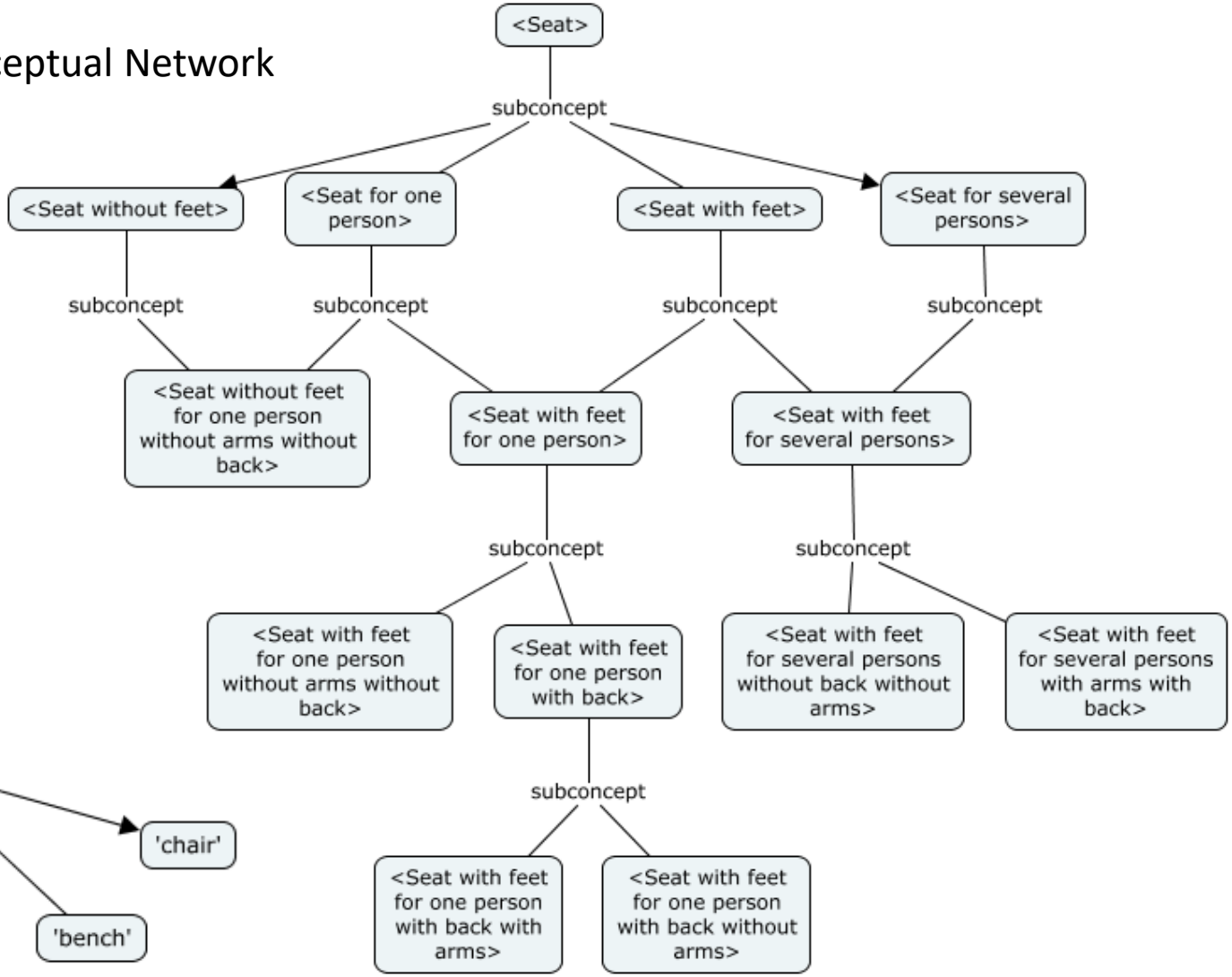
rdfs:seeAlso: https://fr.wikipedia.org/wiki/Chaise
skos:exactMatch: http://vocab.getty.edu/page/aat/300037772

Web reference: Le grand salon de l'Impératrice au château de Fontainebleau du temps de l'impératrice Eugénie
Illustration: Chaises estampillées Georges Jacob, grand cabinet du Dauphin, château de Versailles.

Objects of this type: 2

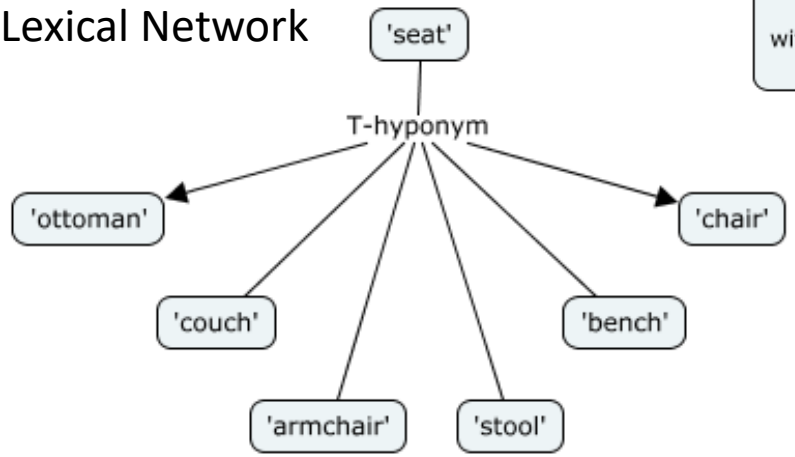
(4) Export: CSV

Conceptual Network



CmapTools

Lexical Network



(4) Export: RDF


```xml
<?xml version="1.0" encoding="utf-8"?>
<!-- Ontoterminology: Ontoterminology of seats -->
<!-- Author: Christophe -->
<!-- Creation date of ontoterminology: 29 juin 2017 -->
<!-- Export date: 25 mai 2020 time: 13:02:51 -->
<!-- Generated by Tedi version: 2.1 - http://christophe-roche.fr/tedi -->

<rdf:RDF xmlns="http://www.ontologia.fr/OTB/Seat#"
    xml:base="http://www.ontologia.fr/OTB/Seat"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:skos="http://www.w3.org/2004/02/skos/core#"
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:xml="http://www.w3.org/XML/1998/namespace"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:vs="http://www.w3.org/2003/06/sw-vocab-status/ns#"
    xmlns:vann="http://purl.org/vocab/vann/">
    <owl:Ontology rdf:about="http://www.ontologia.fr/OTB/Seat.rdf">
        <dc:title>Ontoterminology of seats</dc:title>
        <dc:description>Ontoterminology of seats considered as things made or used for sitt
        on, such as a chair or stool.</dc:description>
        <dc:issued rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2017-6-29</dc:issue
        <dc:modified rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2020-5-25</dc:mod
        <dc:creator>Christophe Roche</dc:creator>
        <dc:publisher>Condillac</dc:publisher>
    </owl:Ontology>

    <!-- Object Properties: -->

    <owl:ObjectProperty rdf:about="#hasPart">
        <owl:inverseOf rdf:resource="#partOf"/>
        <rdfs:domain rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
        <rdfs:range rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
    </owl:ObjectProperty>
```

```
==========================================
              Core Vocabularies for RDF Export
==========================================

Concepts:
------------
- owl:Class                  - owl:DatatypeProperty
- owl:NamedIndividual        - owl:ObjectProperty
- owl:Ontology

Object Properties:
----------------------
- foaf:depiction             - owl:sameAs
- rdf:type                   - rdfs:comment
- rdfs:label                 - rdfs:seeAlso
- rdfs:subClassOf            - skos:altLabel
- skos:broader               - skos:broadMatch
- skos:closeMatch            - skos:definition
- skos:exactMatch            - skos:narrowMatch
- skos:note                  - skos:prefLabel
- skos:related               - skos:relatedMatch
- skos:scopeNote
```

(4) Export: RDF

Concept

```
<owl:Class rdf:about="#Seat_with_feet_for_one_person_with_back_without_arms">
    <skos:prefLabel xml:lang="fr">chaise</skos:prefLabel>
    <skos:definition xml:lang="fr">Siège avec dossier, pour une personne, avec pieds, sans bras.</skos:definition>
    <skos:note xml:lang="fr">Siège à dossier et généralement sans bras. Source : TLFi 31052017</skos:note>
    <skos:example xml:lang="fr">"Les cafés à l'entour avaient depuis longtemps couché sur leurs tables de marbre leurs
    chaises tendrement rabotées par des derrières peu soucieux de voyages. QUENEAU, Pierrot mon ami, 1942, p. 199." Source
    : TLFi 31052017</skos:example>
    <skos:altLabel xml:lang="fr">chaise d'intérieur</skos:altLabel>
    <skos:definition xml:lang="fr">Siège avec dossier, pour une personne, avec pieds, sans bras.</skos:definition>
    <skos:prefLabel xml:lang="en">chair</skos:prefLabel>
    <skos:definition xml:lang="en">Seat for one person, with back and legs, without arms.</skos:definition>
    <skos:note xml:lang="en">"A chair is a piece of furniture for one person to sit on. Chairs have a back and four legs."
https:www.collinsdictionary.comdictionaryenglishchair
2020 05 17</skos:note>
    <skos:example xml:lang="en">‘Cafe tables and chairs face a bench seat below a lowered ceiling that curves for an
    atmospheric effect.’
https:www.lexico.comendefinitionchair
2020 05 25</skos:example>
    <skos:prefLabel xml:lang="gr">καρέκλα</skos:prefLabel>
    <skos:definition xml:lang="gr">Κάθισμα για ένα άτομο, χωρίς μπράτσα, με πλάτη και πόδια. </skos:definition>
    <skos:broader rdf:resource="#Seat_with_feet_for_one_person_with_back"/>
    <rdfs:subClassOf rdf:resource="#Seat_with_feet_for_one_person_with_back"/>
    <rdfs:subClassOf rdf:resource="#without_arms"/>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="hasPart"/>
            <owl:someValuesFrom rdf:resource="#Back"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <foaf:depiction rdf:resource="https://upload.wikimedia.org/wikipedia/commons/thumb/e/e0/Ch%C3%A2teau_de_Versailles%2C_a
    ppartement_du_Dauphin%2C_grand_cabinet_du_Dauphin%2C_chaises.jpg/800px-Ch%C3%A2teau_de_Versailles%2C_appartement_du_Dau
    phin%2C_grand_cabinet_du_Dauphin%2C_chaises.jpg"/>
    <skos:exactMatch rdf:resource="http://vocab.getty.edu/page/aat/300037772"/>
    <rdfs:seeAlso rdf:resource="https://fr.wikipedia.org/wiki/Chaise"/>
</owl:Class>
```

(4) Export: RDF

Individual

```
<owl:NamedIndividual rdf:about="#Chaise_T_507_C">
    <skos:prefLabel>Chaise T 507 C</skos:prefLabel>
    <rdf:type rdf:resource="#Seat_with_feet_for_one_person_with_back_without_arms"/>
    <foaf:depiction rdf:resource="https://journals.openedition.org/crcv/docannexe/
    image/13475/img-13-small480.jpg"/>
    <rdfs:seeAlso rdf:resource="https://journals.openedition.org/crcv/13475?lang=en"/>
</owl:NamedIndividual>
```

(5) Ontoterminology and W3C

1

http://ontologia.fr/OTB/seat.rdf

RDF Export

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Ontoterminology: Ontoterminology of seats -->
<!-- Author: Christophe -->
<!-- Creation date of ontoterminology: 29 juin 2017 -->
<!-- Export date: 25 mai 2020 time: 13:02:51 -->
<!-- Generated by Tedi version: 2.1 - http://christophe-roche.fr/tedi -->

<rdf:RDF xmlns="http://www.ontologia.fr/OTB/Seat#"
    xml:base="http://www.ontologia.fr/OTB/Seat"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:skos="http://www.w3.org/2004/02/skos/core#"
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:xml="http://www.w3.org/XML/1998/namespace"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:vs="http://www.w3.org/2003/06/sw-vocab-status/ns#"
    xmlns:vann="http://purl.org/vocab/vann/">
    <owl:Ontology rdf:about="http://www.ontologia.fr/OTB/Seat.rdf">
        <dc:title>Ontoterminology of seats</dc:title>
        <dc:description>Ontoterminology of seats considered as things made or used for sitting
        on, such as a chair or stool.</dc:description>
        <dc:issued rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2017-6-29</dc:issued>
        <dc:modified rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2020-5-25</dc:modified>
        <dc:creator>Christophe Roche</dc:creator>
        <dc:publisher>Condillac</dc:publisher>
    </owl:Ontology>

    <!-- Object Properties: -->

    <owl:ObjectProperty rdf:about="#hasPart">
        <owl:inverseOf rdf:resource="#partOf"/>
        <rdfs:domain rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
        <rdfs:range rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
    </owl:ObjectProperty>
```

2

SPARQL

3

Protégé

Querying in SPARQL

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

SELECT ?name ?definition
❶ FROM <http://ontologia.fr/OTB/krater.rdf>

WHERE {
        ?concept rdf:type owl:Class.
        ?concept skos:prefLabel ?name.
        ?concept skos:definition ?definition.
        FILTER (lang(?name) = 'en')
        FILTER (lang(?definition) = 'en')
 }

ORDER BY ?name
LIMIT 100
```

https://data.bnf.fr/current/sparql.html

Editeur SPARQL de data.bnf.fr

Graphe par défaut (IRI)

❷ http://ontologia.fr/OTB/seat.rdf

Requête

```
1  PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2  PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
3
4  SELECT ?name ?definition
5  FROM <http://ontologia.fr/OTB/krater.rdf>
6
7  WHERE {
8      ?concept rdf:type owl:Class.
9      ?concept skos:prefLabel ?name.
10     ?concept skos:definition ?definition.
11     FILTER (lang(?name) = 'en')
12     FILTER (lang(?definition) = 'en')
13   }
14
15 ORDER BY ?name
16 LIMIT 100
```

❶ or ❷

❷ Sponging: Retrieve remote RDF data for all missing source graphs

Querying in SPARQL

name	definition
"armchair"@en	"Seat for one person, with arms, back and feet, "@en
"bench"@en	"Seat without back or arms, for more than one persons, with feet, "@en
"chair"@en	"Seat for one person, with back and legs, without arms."@en
"couch"@en	"Seat formore than one people, with arms, back and feet. "@en
"ottoman"@en	"Seat without back or armrsets, for one person, without feet, which usually serves as foot rest or box, with the seat part hinged onto the lower part."@en
"seat"@en	"Piece of furniture designed for seating on."@en
"stool"@en	"Seat for one person, with feet, wihout arm and back. "@en

Querying in SPARQL

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT ?label ?type ?img

WHERE {
        ?object rdf:type owl:NamedIndividual.
        ?object skos:prefLabel ?label.
        ?object rdf:type ?concept.
        ?concept skos:prefLabel ?type.
        ?object foaf:depiction ?img.
        FILTER (lang(?type)='en')
 }

ORDER BY ?label
LIMIT 100
```

https://data.bnf.fr/current/sparql.html

Graphe par défaut (IRI)

http://ontologia.fr/OTB/seat.rdf

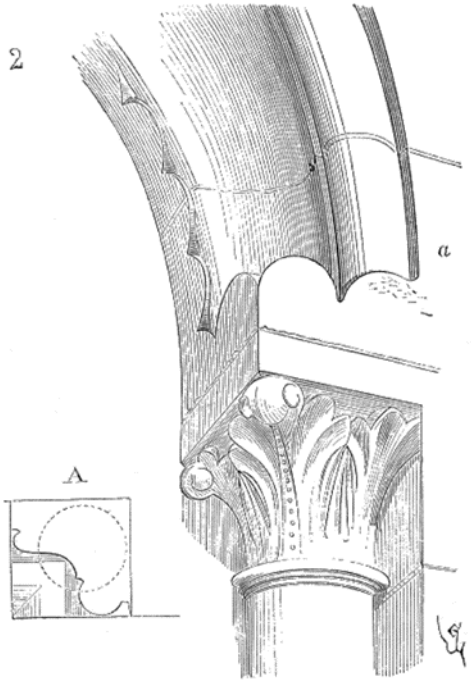
Requête

```
1  PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2  PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
3  PREFIX foaf: <http://xmlns.com/foaf/0.1/>
4
5  SELECT ?label ?type ?img
6
7  WHERE {
8      ?object rdf:type owl:NamedIndividual.
9      ?object skos:prefLabel ?label.
10     ?object rdf:type ?concept.
11     ?concept skos:prefLabel ?type.
12     ?object foaf:depiction ?img.
13     FILTER (lang(?type)='en')
14   }
15 ORDER BY ?label
16 LIMIT 100
```

Sponging: Retrieve remote RDF data for all missing source grap

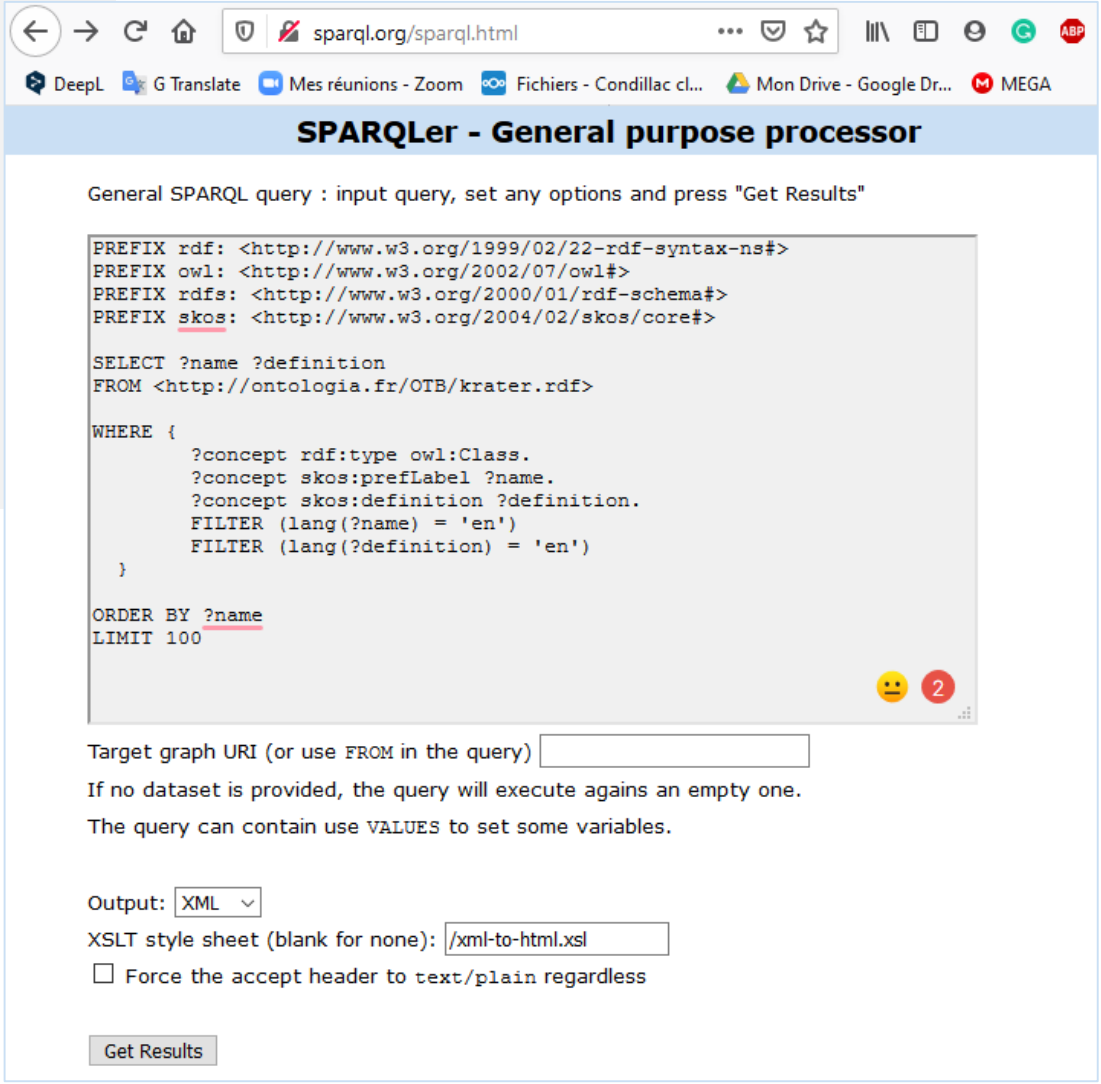
Querying in SPARQL

label	type	img
"Canapé GMT 21269"	"couch"@en	https://journals.openedition.org/crcv/docannexe/image/13475/img-3-small480.jpg
"Canapé T 504 C"	"couch"@en	https://journals.openedition.org/crcv/docannexe/image/13475/img-10-small480.jpg
"Chaise F 263"	"chair"@en	https://journals.openedition.org/crcv/docannexe/image/13475/img-6-small580.jpg
"Chaise T 507 C"	"chair"@en	https://journals.openedition.org/crcv/docannexe/image/13475/img-13-small480.jpg
"Fauteuil F 914 C"	"armchair"@en	https://journals.openedition.org/crcv/docannexe/image/13475/img-5.jpg
"Fauteuil T 505 C"	"armchair"@en	https://journals.openedition.org/crcv/docannexe/image/13475/img-11.jpg
"Fauteuil T 506 C"	"armchair"@en	https://journals.openedition.org/crcv/docannexe/image/13475/img-12.jpg
"Tabouret GMT 1414/6"	"stool"@en	https://journals.openedition.org/crcv/docannexe/image/13475/img-9-small480.jpg
"Tabouret S 574"	"stool"@en	https://journals.openedition.org/crcv/docannexe/image/13475/img-7-small580.jpg



```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

SELECT ?name ?definition
FROM <http://ontologia.fr/OTB/krater.rdf>

WHERE {
        ?concept rdf:type owl:Class.
        ?concept skos:prefLabel ?name.
        ?concept skos:definition ?definition.
        FILTER (lang(?name) = 'en')
        FILTER (lang(?definition) = 'en')
 }

ORDER BY ?name
LIMIT 100
```

http://sparql.org/sparql.html



Selected References

Mind the Gap: Ontology Authoring for Humanists

C. Roche M. Papadopoulou. 1st International Workshop for Digital Humanities and their Social Analysis (WODHSA)- Episode V: The Styrian Autumn of Ontology, September 23-25, a Workshop hosted by Joint Ontology Workshops, Medical University of Graz (Austria), September 23-25, 2019

Building ontology-based dictionaries for Greek material culture terms

M. Papadopoulou & C. Roche. 1st International Workshop on Open Data and Ontologies for Cultural Heritage. Rome, Italy, June 3, 2019

Twinning Classics and A.I.: Building the new generation of ontology-based lexicographical tools and resources for Humanists on the Semantic Web

M. Papadopoulou, C. Roche. In S. Krauwer, D. Fišer (eds.) – Twin Talks at DHN 2018 – Understanding Collaboration in Digital Humanities. Open access, University of Copenhagen, 5th March 2019

Ontologization of Terminology. A worked example from the domain of ancient Greek dress

M. Papadopoulou, C. Roche. AIDAinformazioni Journal, number 1-2/2018, volume XXXVI

Ontological definition – abstract

C.Roche. Handbook of Terminology, Volume 1, John Benjamins Publishing, 2015, pp.128-152

Ontoterminology: How to unify terminology and ontology into a single paradigm

C.Roche. LREC 2012, Eighth international conference on Language Resources and Evaluation, Istanbul (Turkey), 21-27 May 2012, pp. 2626-2630

Any Comments or Remarks?

send an email to: roche@univ-savoie.fr

http://ontoterminology.com/