
Was sind Ontologie-Editoren?

Kurzeinführung Protégé

Gliederung

- ▶ **Ontologie Editoren- allgemein**
 - ▶ warum nutzen
 - ▶ wofür nutzen
 - ▶ Probleme
- ▶ **Marktlage**
- ▶ **Einführung in die praktische Arbeit mit Protégé**
- ▶ **Protégé Praxis**



Ontologie Editoren

- ▶ Software mit der Ontologien erstellt werden können
- ▶ Editoren bieten Tools die bei der Erstellung unterstützen
- ▶ Dienen zur Visualisierung und Katalogisierung von Wissen

Ontologie Editoren im Einsatz für...

- ▶ **Informationsontologien**
 - ▶ beschreiben Informationsquellen und Metaeigenschaften
- ▶ **Terminologische Ontologien**
 - ▶ beschreiben wie Menschen über eine Domäne reden
 - ▶ bilden Verbindung zu Thesauri und Information Retrieval
- ▶ **Wissensontologien**
 - ▶ detaillierte und formal „saubere“ Wissensmodelle
 - ▶ Expertensysteme
- ▶ **Unternehmensontologien**
 - ▶ Organisationsstruktur
 - ▶ Kontext → Prozesse und Personen

-
- ▶ Einsatz Ontologie Editoren

Probleme beim Einsatz von Ontologie Editoren

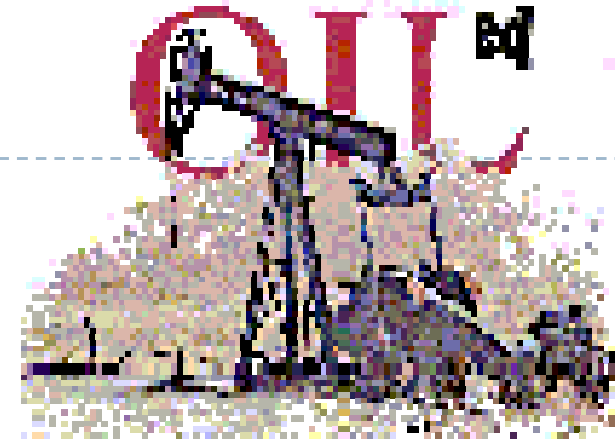
- ▶ **Gemeinsames Verständnis schaffen**
- ▶ **hoher Aufwand für Verständnis**
- ▶ **Vorbedingung für Auswahl von Kategorien**
 - ▶ kann für andere umständlich/unlogisch erscheinen

Welche Ontologie Editoren gibt es?

- ▶ großes Angebot an Ontologie Editoren vorhanden
- ▶ exemplarische Vorstellung
 - ▶ kostenloser Ontologie Editor: OilEd
 - ▶ kommerzieller Ontologie Editor: TopBraid Composer

OilEd

- ▶ entwickelt an der Universität Manchester
 - ▶ Information Management Group
 - ▶ veröffentlicht: 31. Oktober 2003
- ▶ wird seit 2004 nicht mehr gepflegt
- ▶ kostenloser Download nach Registrierung



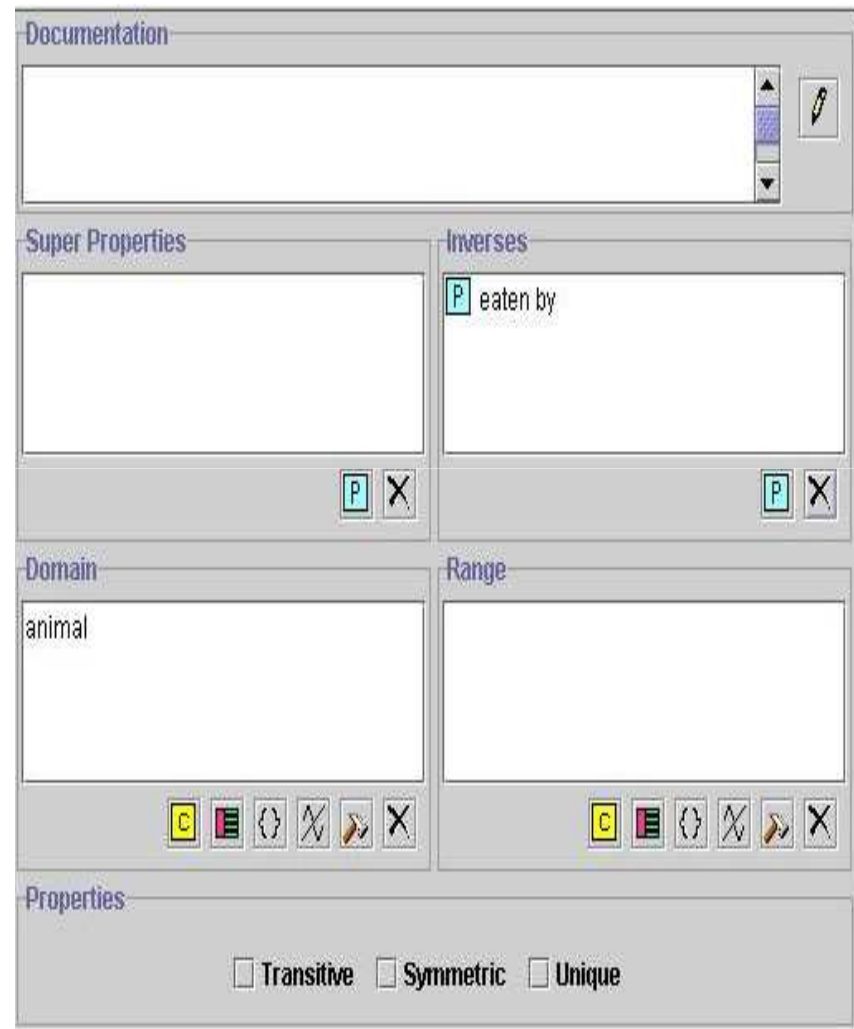
Quelle: <http://www.webservicesummit.com/images/oiled.gif>

OilEd

- ▶ Sprachen: DAML und OIL
- ▶ grafische Darstellung nur mit OilViz möglich
- ▶ RDF und Uris möglich
- ▶ begrenzte Synonym Anzahl
- ▶ begrenzter Platz für Namen

- ▶ Kein Lexikalischer Support
- ▶ Kein Extrahieren möglich

- ▶ fehlender Support



Quelle: <http://www.cs.man.ac.uk/~horrocks/Teaching/cs646/OilEdTutorial-Ver1.1/property-editor.jpg>

- ▶ OilEd

TopBraid Composer



- ▶ kommerzieller Editor der Firma TopQuadrat
 - ▶ erste Veröffentlichung 2006
 - ▶ Version 2.0 Februar 2007
- ▶ basiert auf Eclipse
- ▶ Form und Farbgebung ähnlich Protégé

TopBraid Composer

- ▶ Beschreibungssprachen: RDF, RDFS, OWL,; Abfragesprache: SPARQL, SWRL (Erweiterung zu OWL)
- ▶ Erweiterungen mit Plugins
- ▶ intuitiver als Protégé
- ▶ 30-tägige Testversion

The screenshot displays the TopBraid Composer interface. On the left, a class hierarchy is shown under 'owl:Thing (135)', including 'College (19)', 'Gender (2)', 'owl:Nothing', 'Person (75)', 'Matriarch (1)', 'Patriarch (1)', 'Profession (34)', 'world:Address', 'world:Area', 'world:Country (5)', and 'world:State'. The main workspace shows a class diagram with 'College', 'Profession', and 'Gender' classes. 'College' has a 'photo' property and a 'ivy-league : boolean' property. 'Person' has properties 'alma-mater : College', 'has-child : Person', 'has-parent : Person', 'photo', 'profession : Profession', 'sex : Gender', and 'birth-year : int'. 'Profession' is connected to 'Person' via the 'profession' property, and 'Gender' is connected via the 'sex' property. At the bottom, a table shows instances of the 'Person' class:

[Resource]	rdfs:type	rdfs:label	rdfs:comm...	first-name	death-yea
person27	Person	Arnold Sch...		Arnold	
person28	Person	Timothy Sh...		Timothy	
person3	Person	Joseph Ken...		Joseph	1944
person30	Person	Linda Potter		Linda	
person31	Person	Mark Shriver		Mark	
person32	Person	Jeannie Ripp		Jeannie	

Quelle: <http://www.topquadrant.com/topbraid/composer/images/TBC-Index2-big.png>

- ▶ TopBraid Composer

Protégé



- ▶ einer der bekanntesten Ontologie Editoren
- ▶ entwickelt von der Stanford University
 - ▶ Stanford Biomedical Informatics Research
- ▶ veröffentlicht 22. Juni 2004
- ▶ frei verfügbar
- ▶ kann durch Plugin Konzept erweitert werden und individuell angepasst werden
- ▶ unterstützt sowohl OWL als auch RDF

Protégé

- ▶ **Unterschiede zu anderen Ontologie Editoren**
 - ▶ Kostenlos
 - ▶ viele Beispiel -Ontologien durch weite Verbreitung
 - ▶ Intuitiv und einfach zu nutzen
 - ▶ Datenbank lädt Frames nur auf Verlangen, nutzt Cache zum Speichern
- ▶ **Plugin- Architektur**
 - ▶ verschiedene Plugins ermöglichen Zuschnitt auf unterschiedliche Aufgaben

Einführung in die praktische Arbeit

▶ Protégé-Frame Editor

- ▶ erlaubt frame-basierte Ontologien zu erstellen
- ▶ basiert auf dem dem Open Knowledge Base Connectivity protocol (OKBC)

▶ Protégé-OWL Editor

- ▶ Erweiterung von Protégé
- ▶ Unterstützt OWL (Web Ontology Language)
- ▶ Höhere Ausdrucksfähigkeiten als RDF
- ▶ Ermöglicht komplizierte logische Verbindungen

▶ Einführung

Versionen

Protégé 3x	Protégé 4 beta
frame- basierte Ontologien möglich	Keine Arbeit mit Frames möglich
OWL 1.0	OWL 2.0
OWL und RDF(S)	OWL framework
viele Plug-Ins vorhanden	Nicht alle Plug-Ins sind auch für Version 4 vorhanden
Multi-User Support	Kein Multi-User Support

→ Besser Version 3, hier stehen mehr Möglichkeiten zur Verfügung als zurzeit bei Version 4

Download

- ▶ Registrierung (nicht notwendig)
- ▶ Downloads unter:
<http://protege.stanford.edu/download/registered.html>
- ▶ Auswahl:
 - ▶ Protégé 4
 - ▶ Protégé 3
 - ▶ Ältere Versionen von Protégé
 - ▶ WebProtégé 0.5 alpha

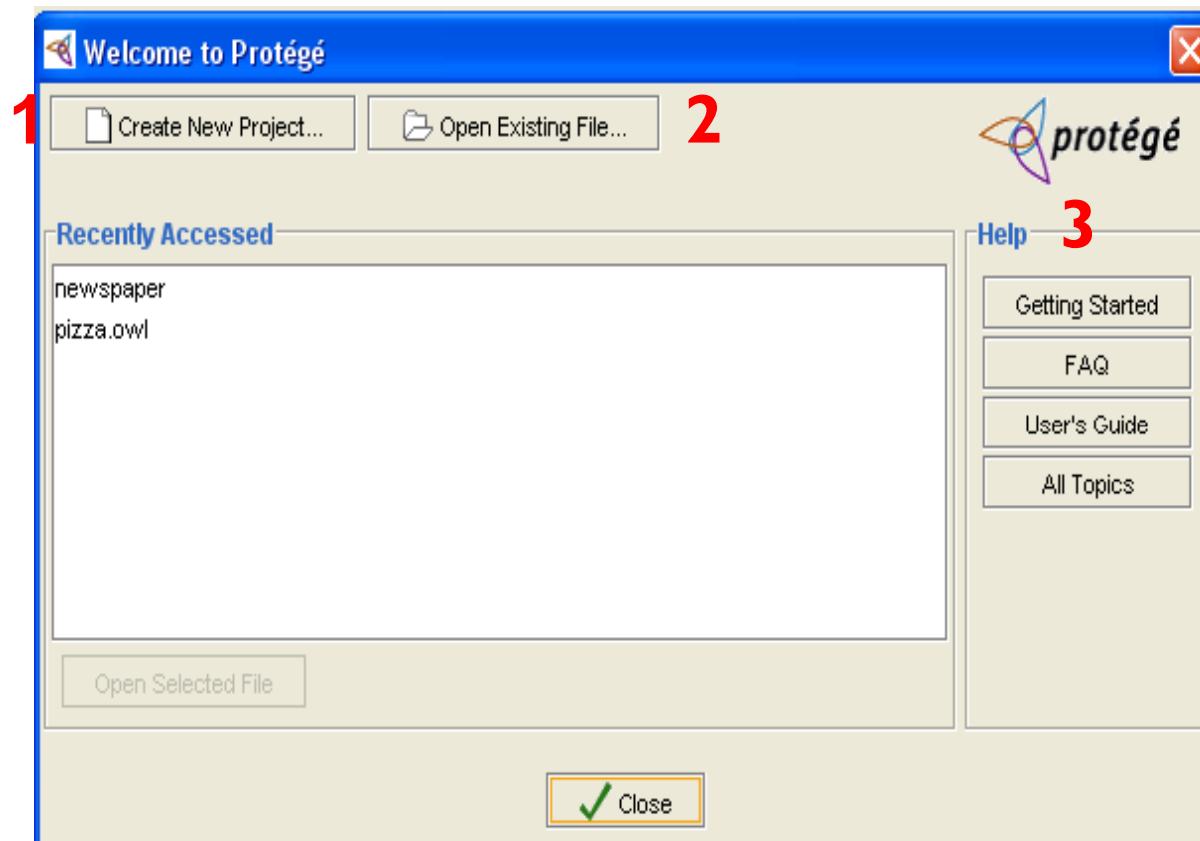
-
- ▶ Einführung

Hilfe und Tutorials

- ▶ **Protégé-OWL Tutorial**
 - ▶ Aufbau einer OWL Ontologie anhand des Pizza –Beispiels
- ▶ **Getting started**
 - ▶ Einführung in Protégé-Frames
- ▶ **Protégé FAQ**
 - ▶ Allgemeine Fragen zu Protégé / Protégé-Frames
 - ▶ FAQ zu Protégé 3 und 4
 - ▶ FAQ zu Speicherformaten
- ▶ **Protégé Wiki**
 - ▶ New Protégé Semantic Media Wiki (seit 2007)

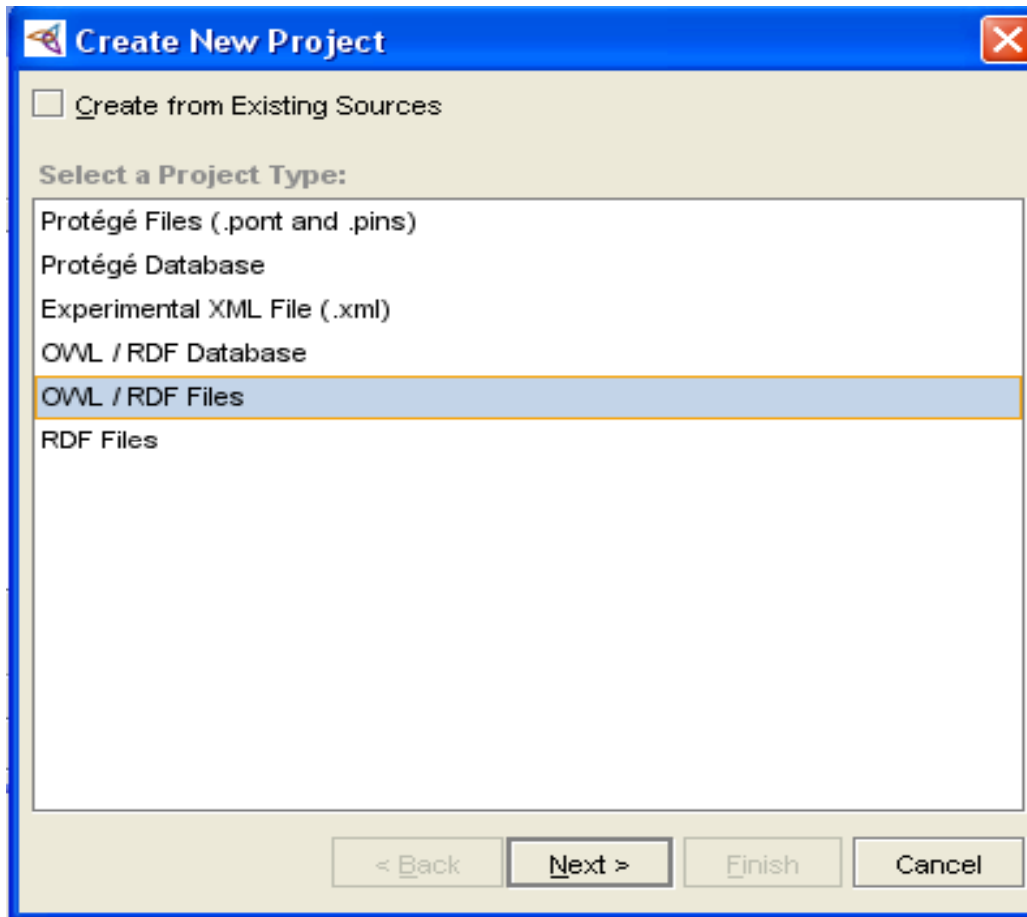
-
- ▶ Einführung

Erste Schritte - neues Projekt anlegen



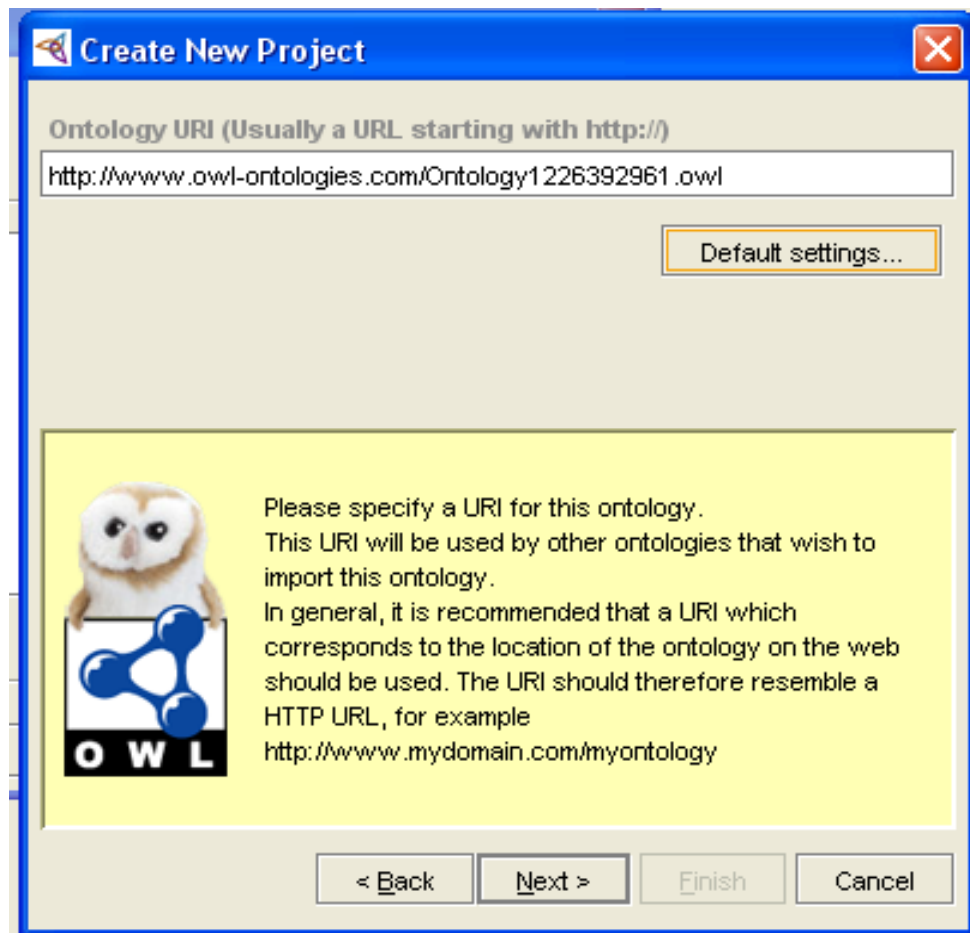
1. Ein neues Projekt anlegen
2. Ein bestehendes Projekt laden
3. Hilfe

Erste Schritte - neues Projekt anlegen



Die Auswahl „OWL/RDF Files“ ermöglicht es mit RDF und OWL zu arbeiten

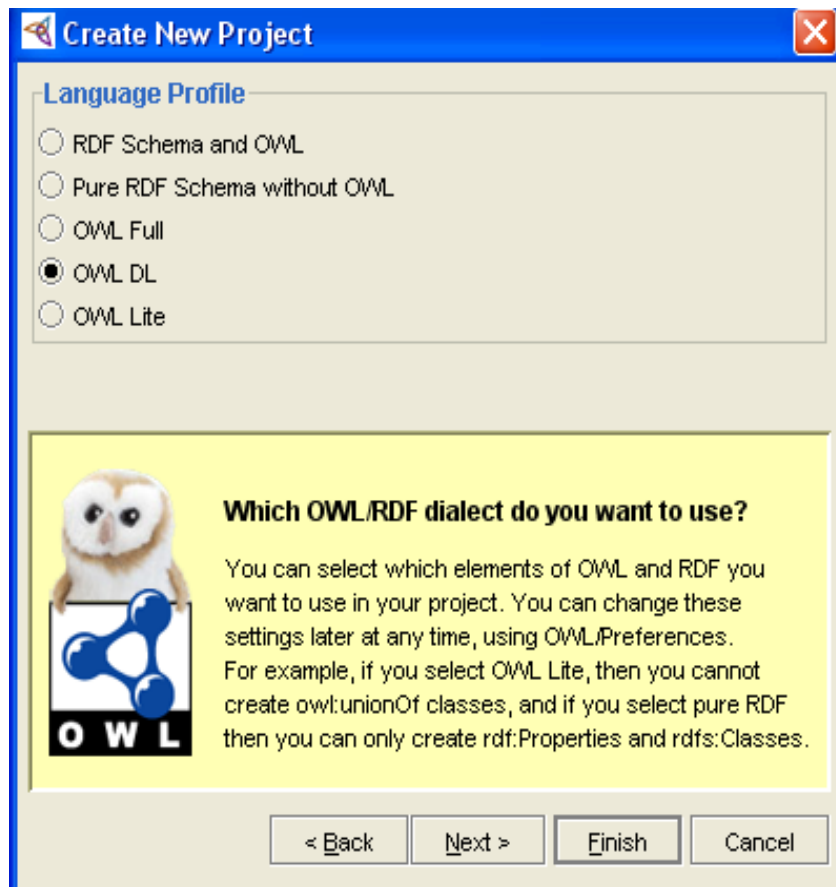
Erste Schritte - neues Projekt anlegen



URI

- ✓ dient der Identifikation der Ontologie
- ✓ sollte den Ort angeben, an welchem die Ontologie später publiziert wird

Erste Schritte - neues Projekt anlegen



Auswahl des OWL/RDF Dialektes:

OWL Lite

- „Light-Version“ von OWL
- z.B. einfache Taxonomien, Hierarchien, Thesauri
- verschiedene Sprachkonstrukte aus OWL DL sind nicht vorhanden

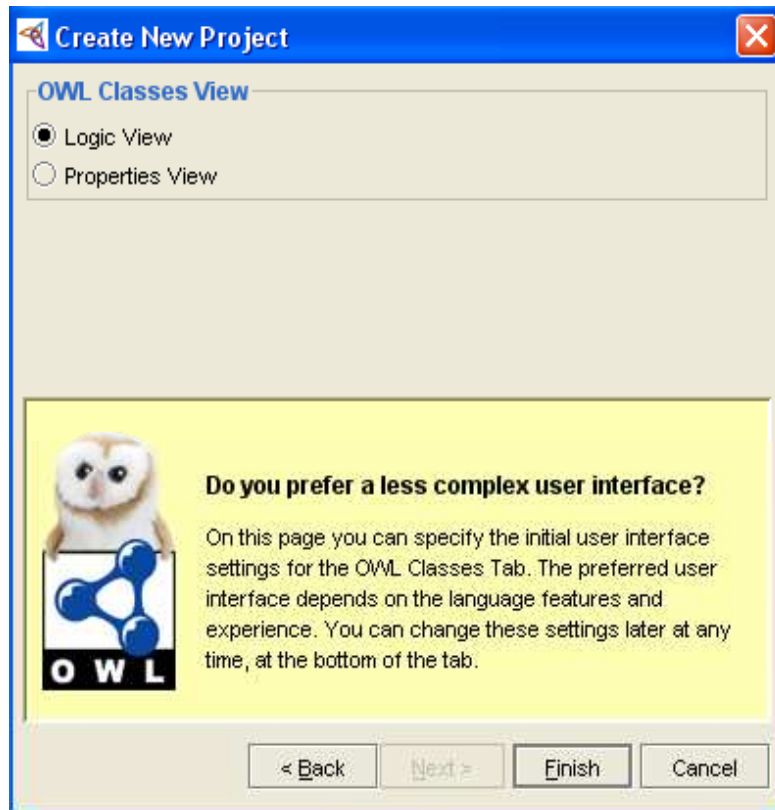
OWL DL

- basiert auf Beschreibungslogik (DL=Description Logic)
- Einschränkungen für den Einsatz von RDFS-Konstrukten

OWL Full

- besteht aus den selben Sprachkonstrukten wie OWL DL
- aber ohne Einschränkungen

Erste Schritte - neues Projekt anlegen



Darstellung der
Benutzeroberfläche

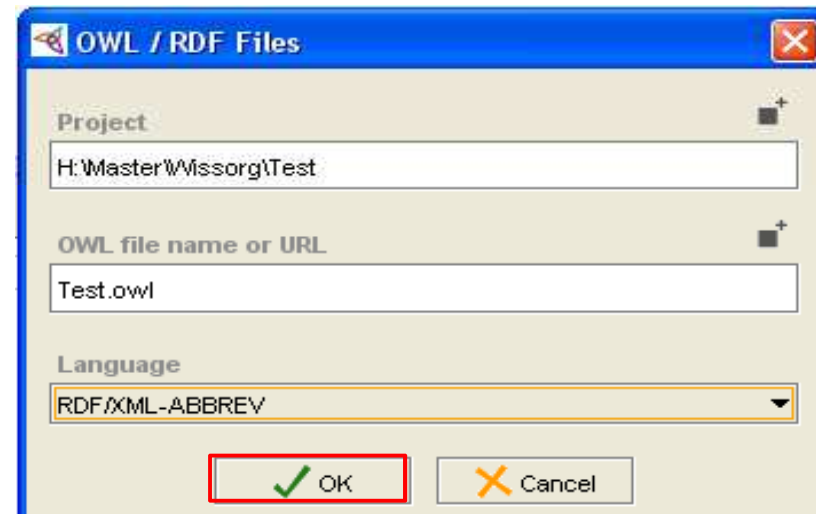
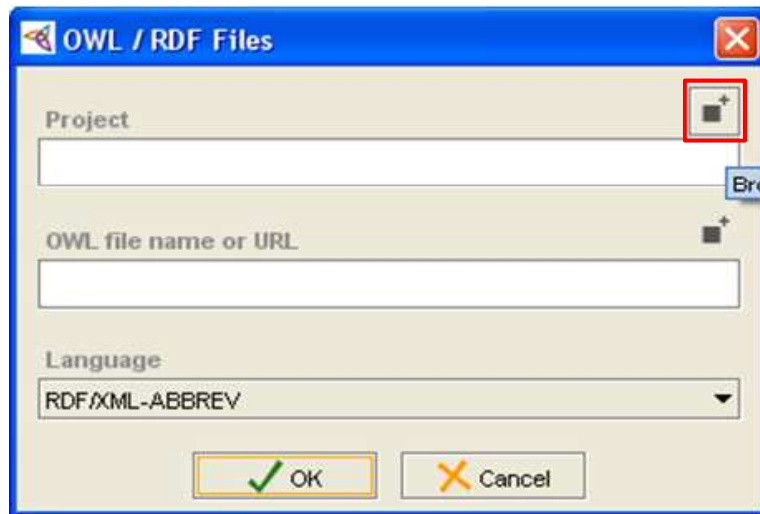
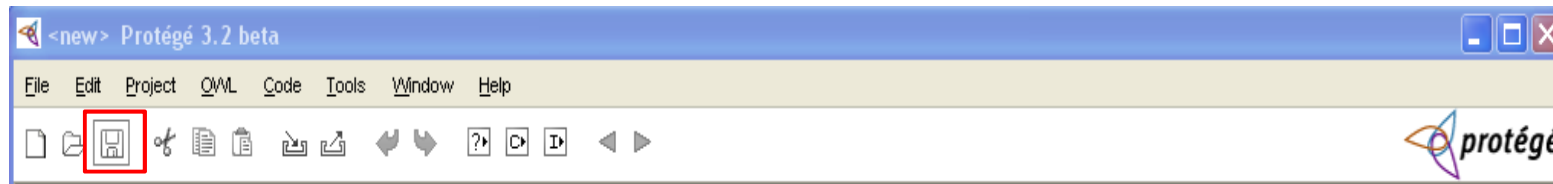
Logic View

- ✓ sehr ausdrucksstark
- ✓ für Anfänger nicht intuitiv verständlich

Properties View

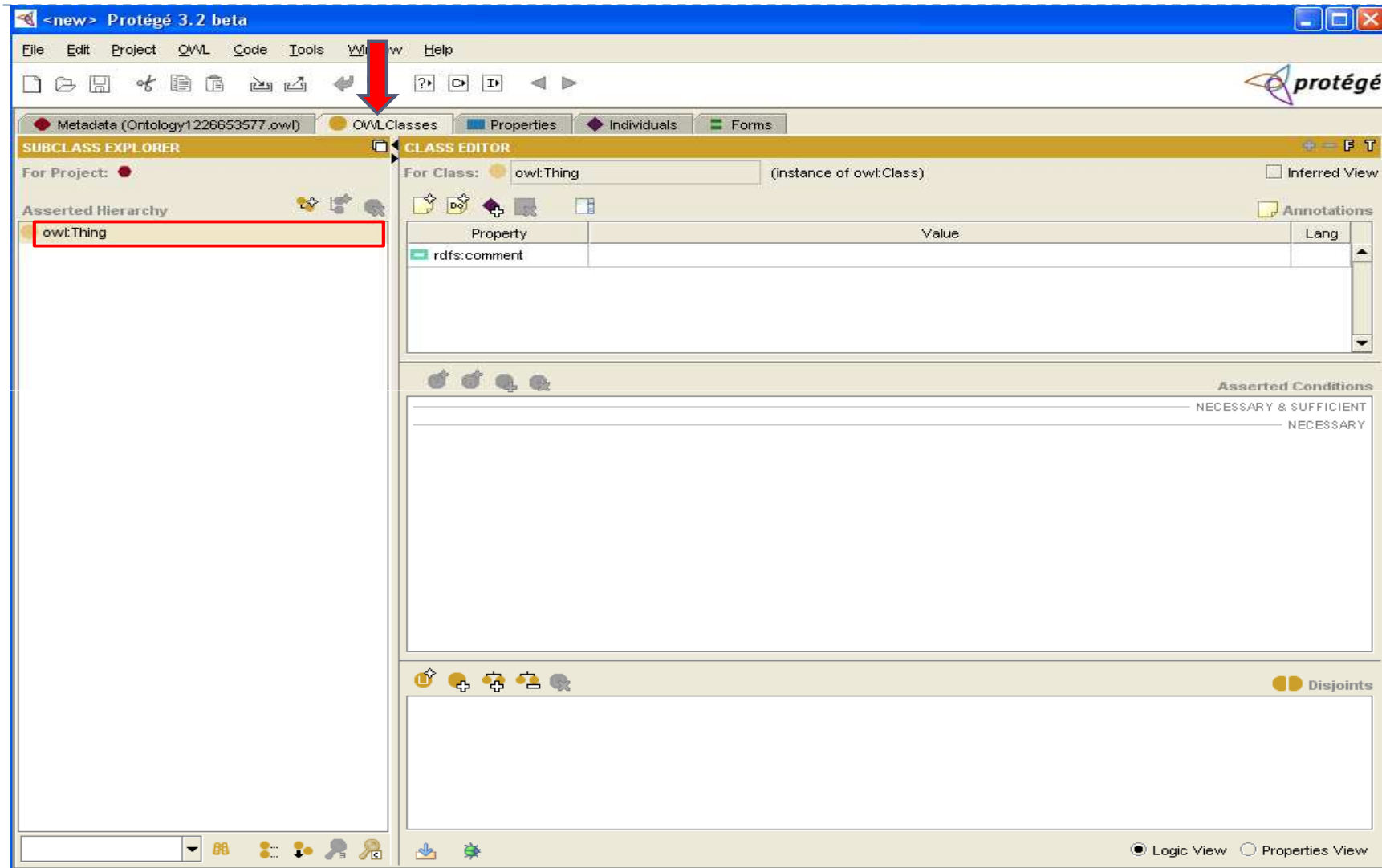
- ✓ einfachere Oberfläche
- ✓ für Anfänger geeigneter

Erste Schritte - ein Projekt speichern

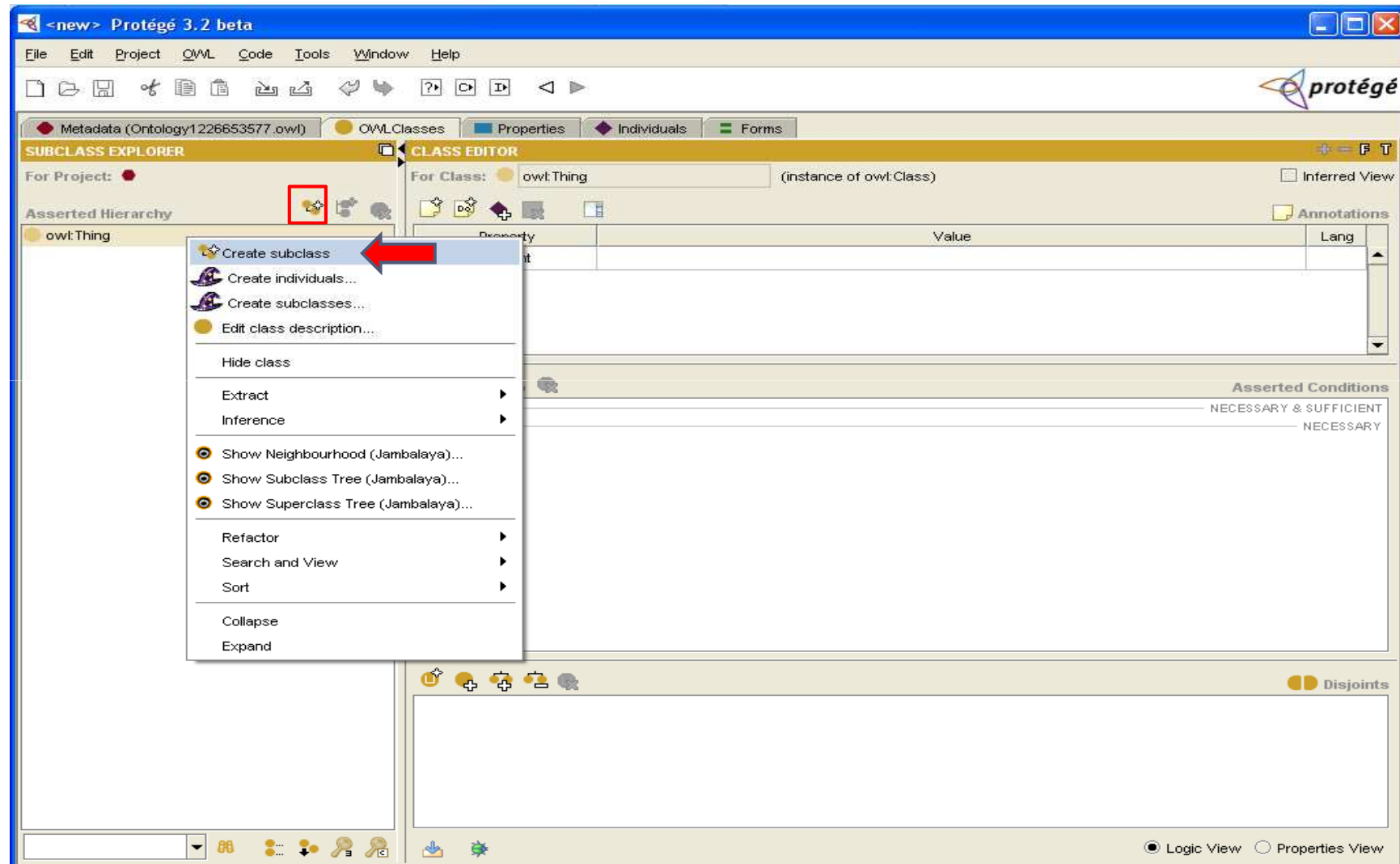


Auswahl des Speicherortes

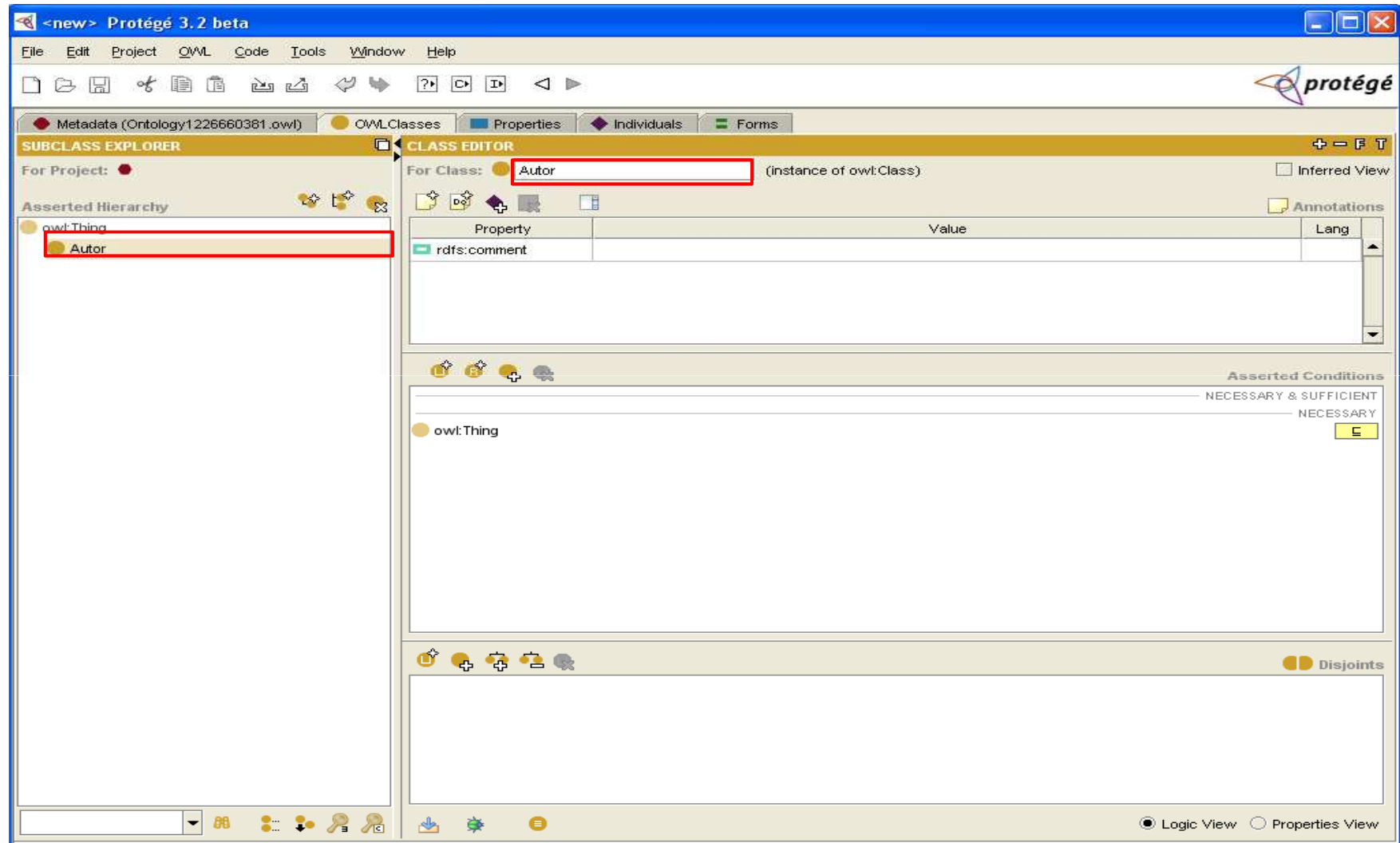
OWL Classes



Klassen und Unterklassen anlegen



Klassen und Unterklassen anlegen



Klassen und Unterklassen anlegen

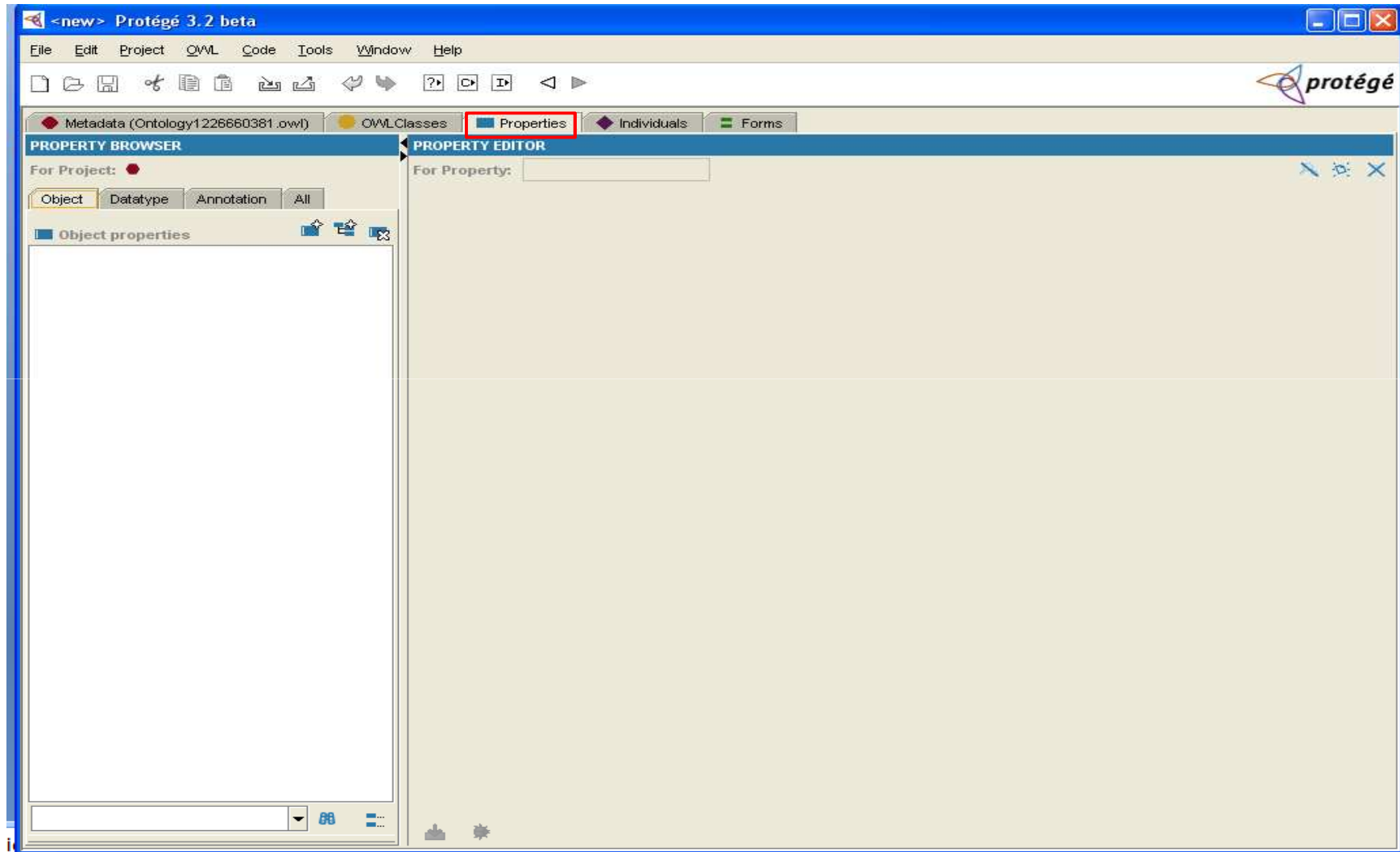
The screenshot displays the Protégé 3.2 beta software interface. The main window is titled "<new> Protégé 3.2 beta" and features a menu bar (File, Edit, Project, OWL, Code, Tools, Window, Help) and a toolbar. The interface is divided into several panes:

- SUBCLASS EXPLORER:** Located on the left, it shows an "Asserted Hierarchy" for the project. The hierarchy starts with `owl:Thing` and includes sub-classes: `Autor` (with sub-classes `Reporter`, `Herausgeber`, and `Kolumnist`) and `Inhalt` (with sub-classes `Artikel` and `Werbung`).
- CLASS EDITOR:** The central pane, currently set for the class `owl:Thing` (instance of `owl:Class`). It includes an "Inferred View" checkbox and an "Annotations" section. Below this is a table for defining properties:

Property	Value	Lang
<code>rdfs:comment</code>		

Below the table are sections for "Asserted Conditions" (with radio buttons for NECESSARY & SUFFICIENT and NECESSARY) and "Disjoints". At the bottom right, there are radio buttons for "Logic View" (selected) and "Properties View".

Properties anlegen



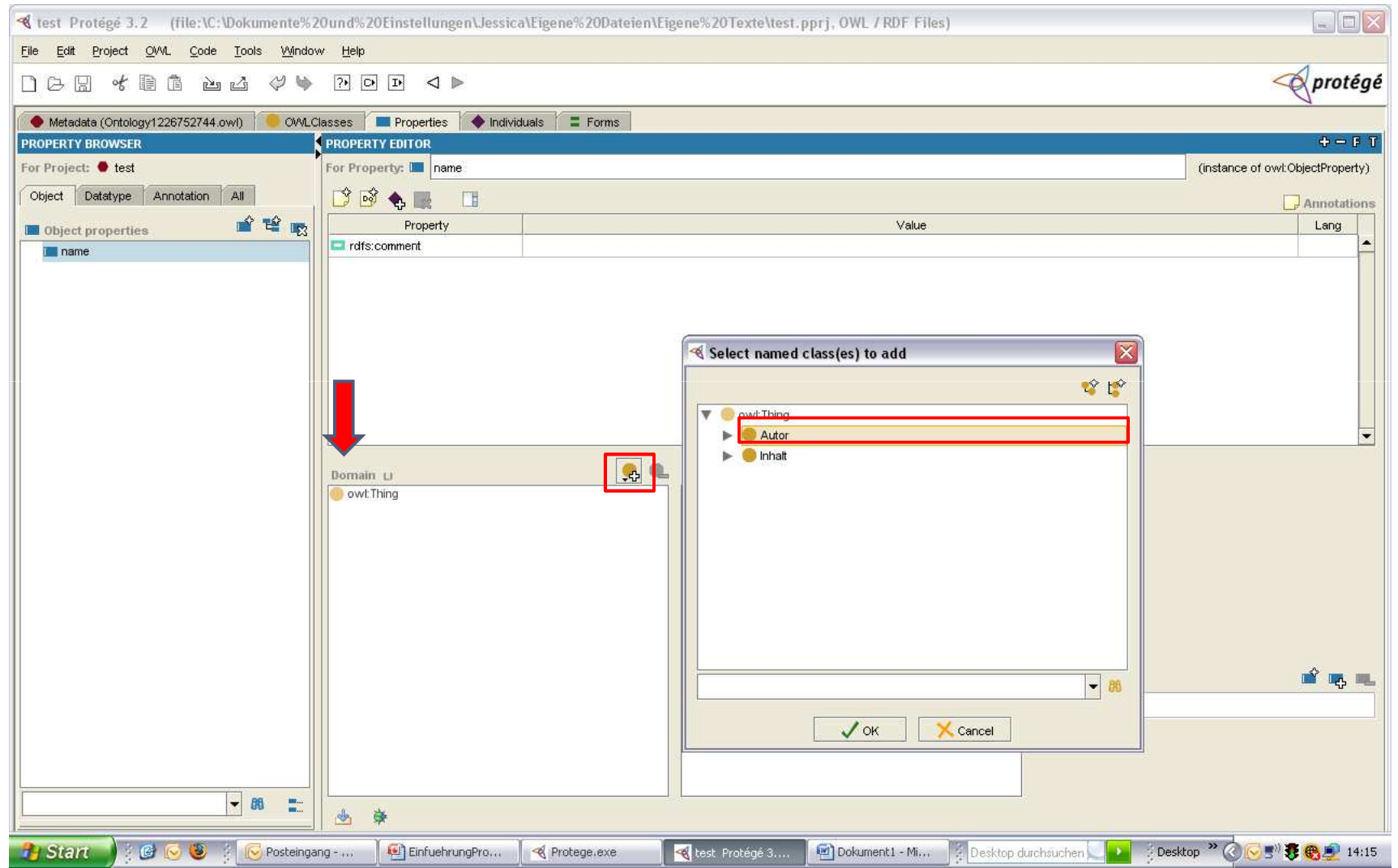
Properties anlegen

The screenshot shows the Protégé 3.2 interface. The main window is titled "test Protégé 3.2 (file:IC:\Dokumente%20und%20Einstellungen\Jessica\Eigene%20Dateien\Eigene%20Texte\test.pprj, OWL / RDF Files)". The interface is divided into several panes:

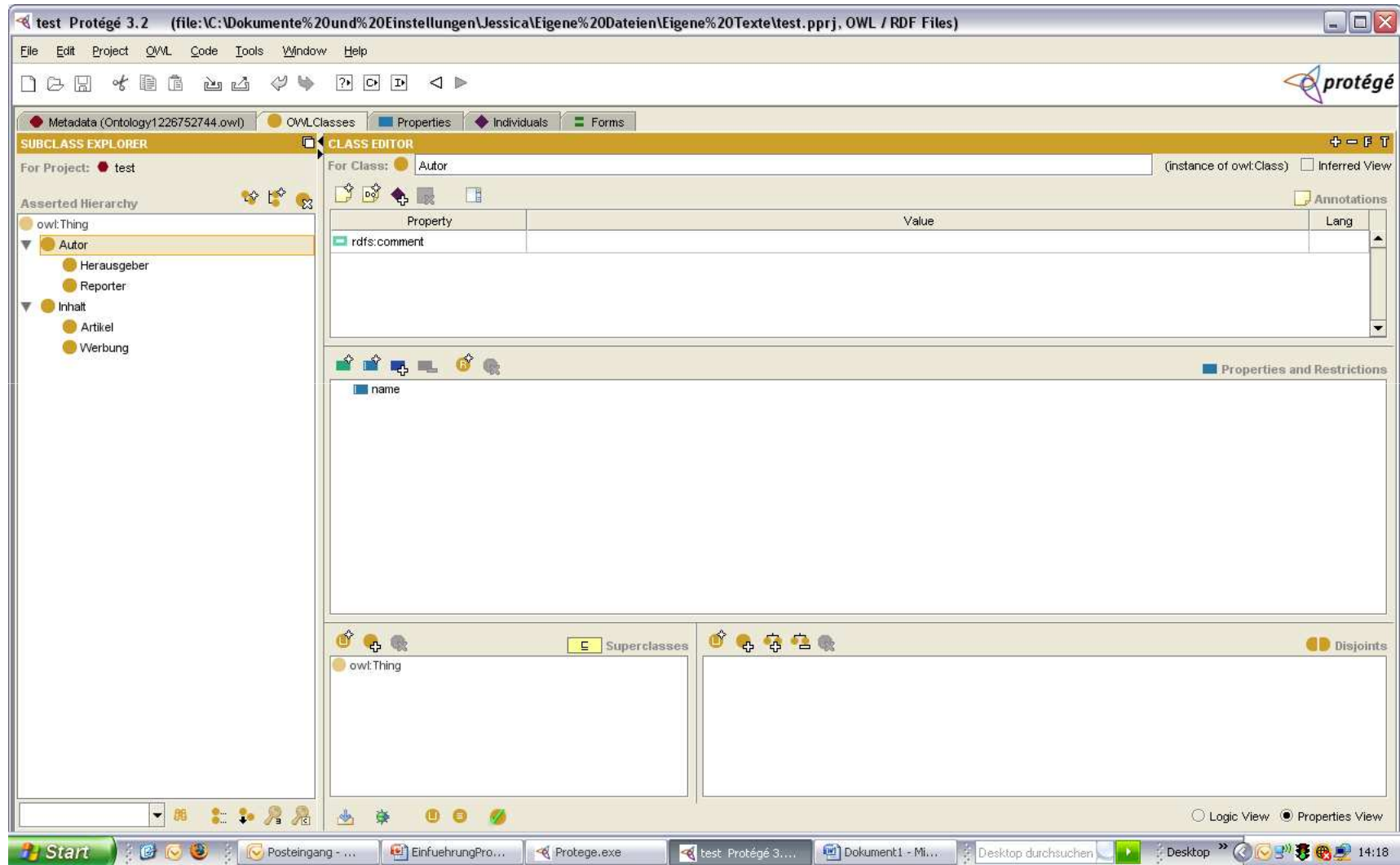
- PROPERTY BROWSER:** Shows the project "test" and a list of "Object properties" with "name" selected and highlighted by a red box.
- PROPERTY EDITOR:** Shows the "For Property:" field with "name" entered, also highlighted by a red box. Below it is a table with columns "Property" and "Value". The table contains one row with "rdfs:comment" in the "Property" column and an empty "Value" column.
- Domain and Range:** Two empty boxes labeled "Domain" and "Range". The "Domain" box has a yellow circle icon and the text "Autor" below it.
- Options:** A list of checkboxes for property characteristics: "Functional", "InverseFunctional", "Symmetric", and "Transitive".
- Inverse:** A checkbox labeled "Inverse" with an empty text field next to it.

The Windows taskbar at the bottom shows the Start button, several icons, and the system tray with the time "14:14".

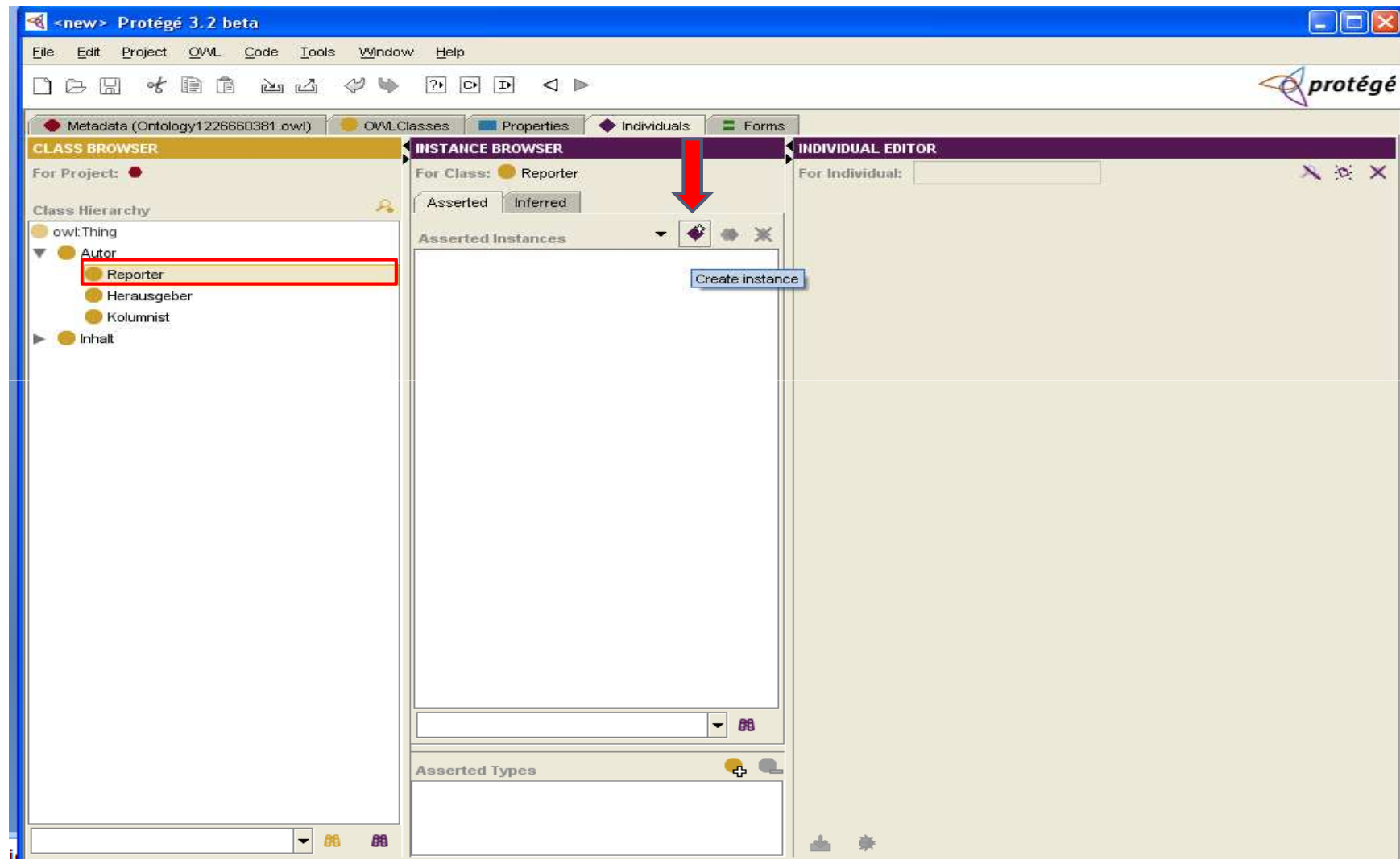
Properties zuordnen - Domain



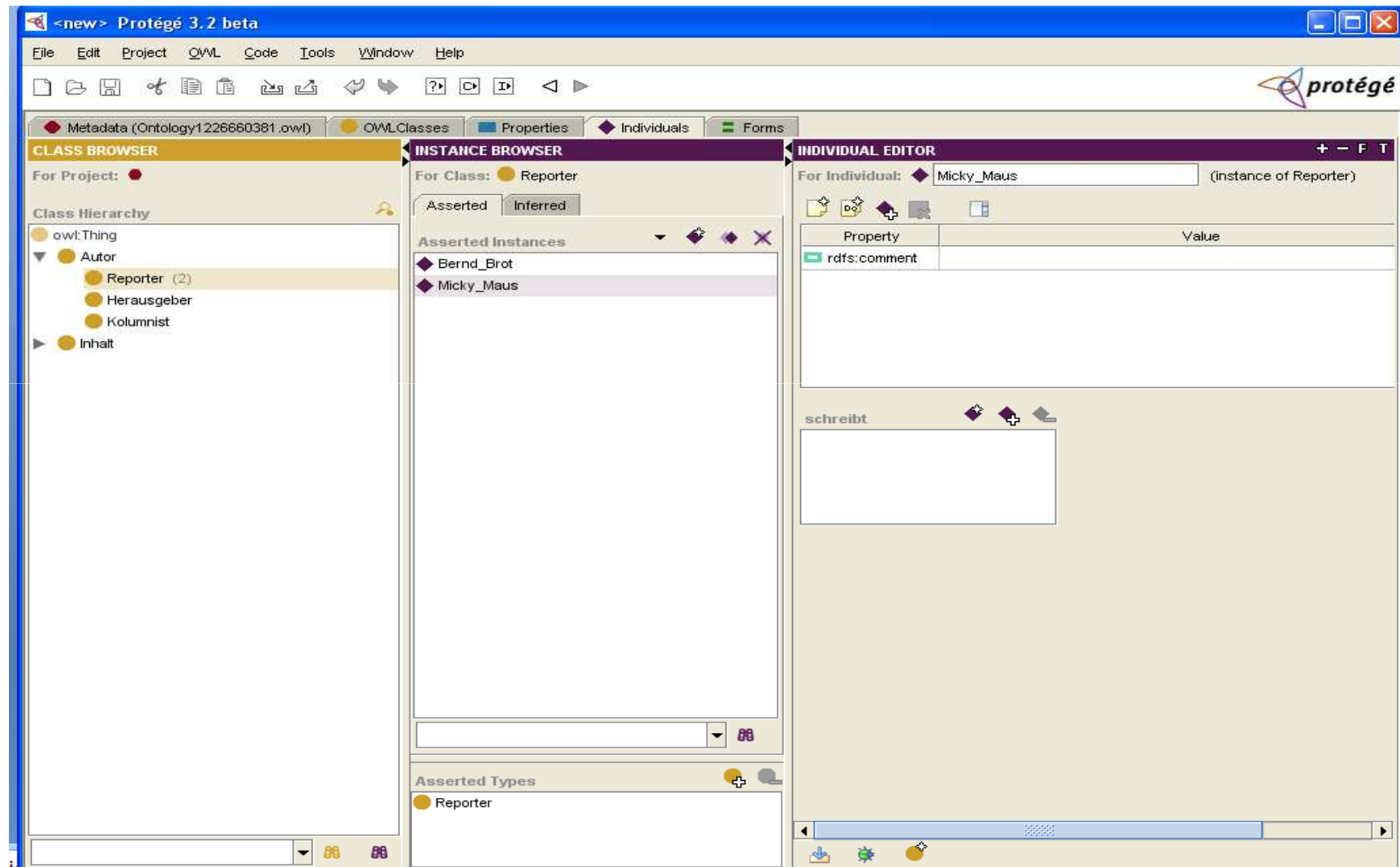
Ansicht „Classes“ nach Properties Zuordnung



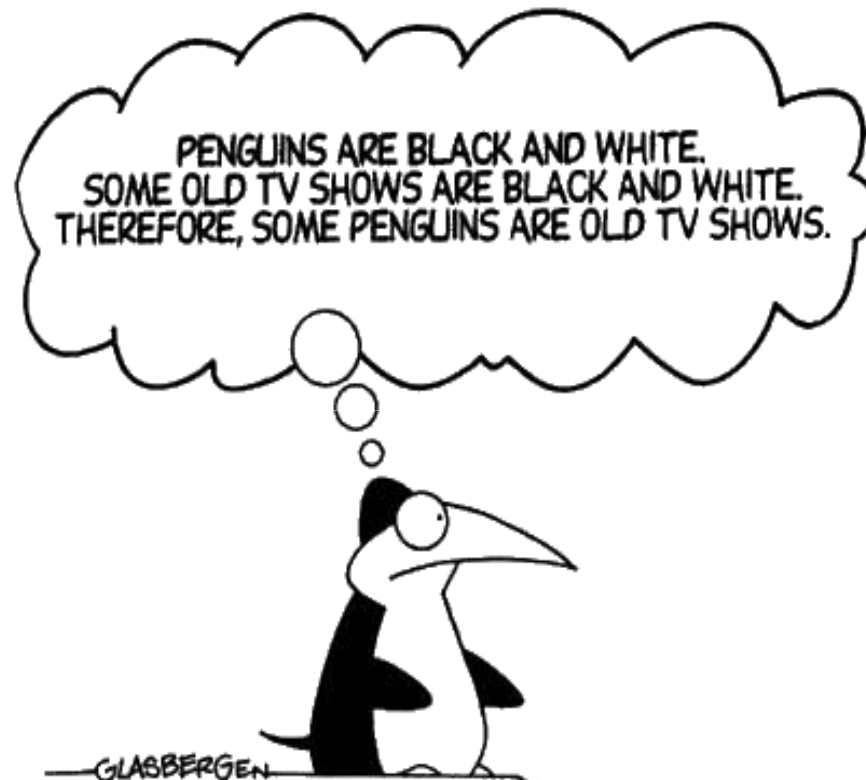
Individuals erstellen



Individuals erstellen



Vielen Dank für eure Aufmerksamkeit!



**Logic: another thing that
penguins aren't very good at.**

Quelle: http://www.cs.man.ac.uk/~liz/images/cartoon_penguin_logic.gif