**The Common Core Ontologies:**

**Guide to Getting Started**

Prepared By: CUBRC, Inc.

Version Date: November 18, 2019

**Introduction:**

This document is intended to help get you started using the Common Core Ontologies (CCO). It consists of 3 sections. Section 1 provides step-by-step instructions for loading the Common Core Ontologies to view or edit using the freely available ontology editor Protégé. The first time loading the ontologies can be frustrating because Protégé needs to build a catalog file so it knows where to locate the file for each imported ontology. Section 2 provides troubleshooting tips and tricks to speed up the process. Finally, Section 3 provides a brief tutorial to get you started browsing the ontologies.

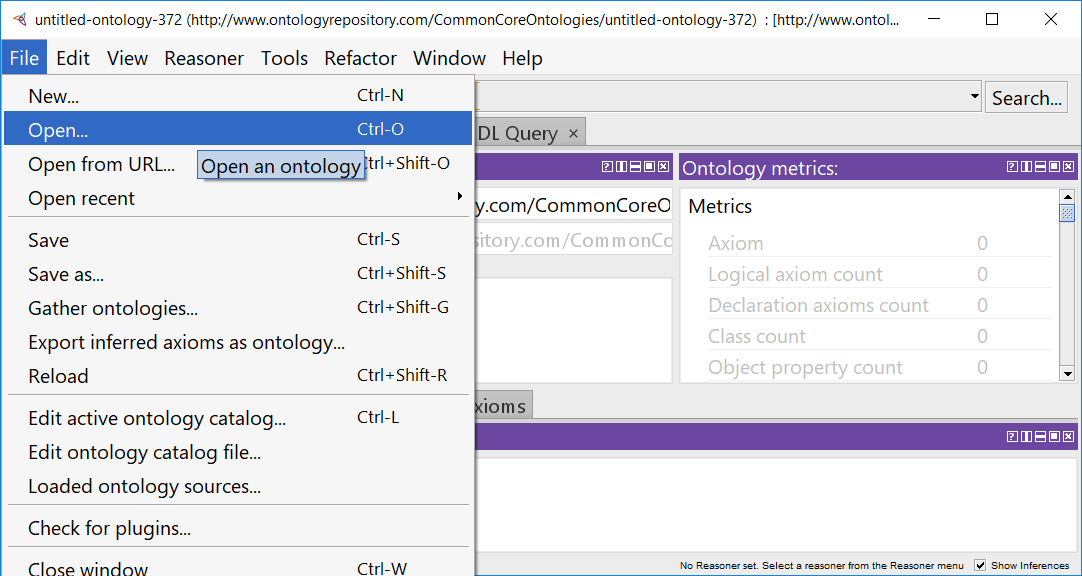
**Section 1: Loading the Ontologies**

**STEP 1:** Download and install the ontology editor Protégé (available for free here: <https://protege.stanford.edu/>)

**STEP 2:** Launch Protégé

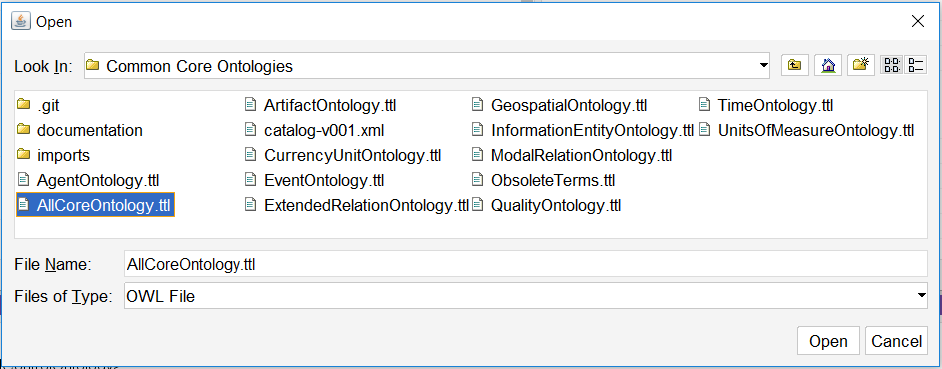
PRO TIP: Turn off your internet connection before performing STEP 3. Refer to [Troubleshooting TIP #1](#TroubleshootingTIP1) for an explanation.

**STEP 3:** Click ‘File’ then ‘Open…’ (or use the keyboard shortcut Ctrl+O)



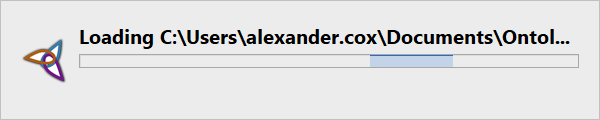
**STEP 4:** Navigate to the folder where the target ontology file is saved. Select the file then click ‘Open’.

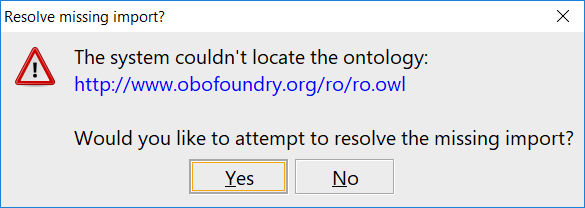
It is recommended that you open AllCoreOntology.ttl first since this file is designed to import all of the Common Core Ontologies (except the Currency Unit Ontology and the Modal Relation Ontology) to facilitate easier browsing of the Common Core.



**STEP 5:** Resolve any missing imports.

If a “Resolve missing import?” window pops up (see second image below), look at the name of the missing file (in this case, ro.owl) and remember the name BEFORE clicking ‘Yes’ as this is the only time the name of the missing import file gets displayed during this process. When ready, click ‘Yes’ then navigate to the appropriate folder and select then open the correct ontology file.



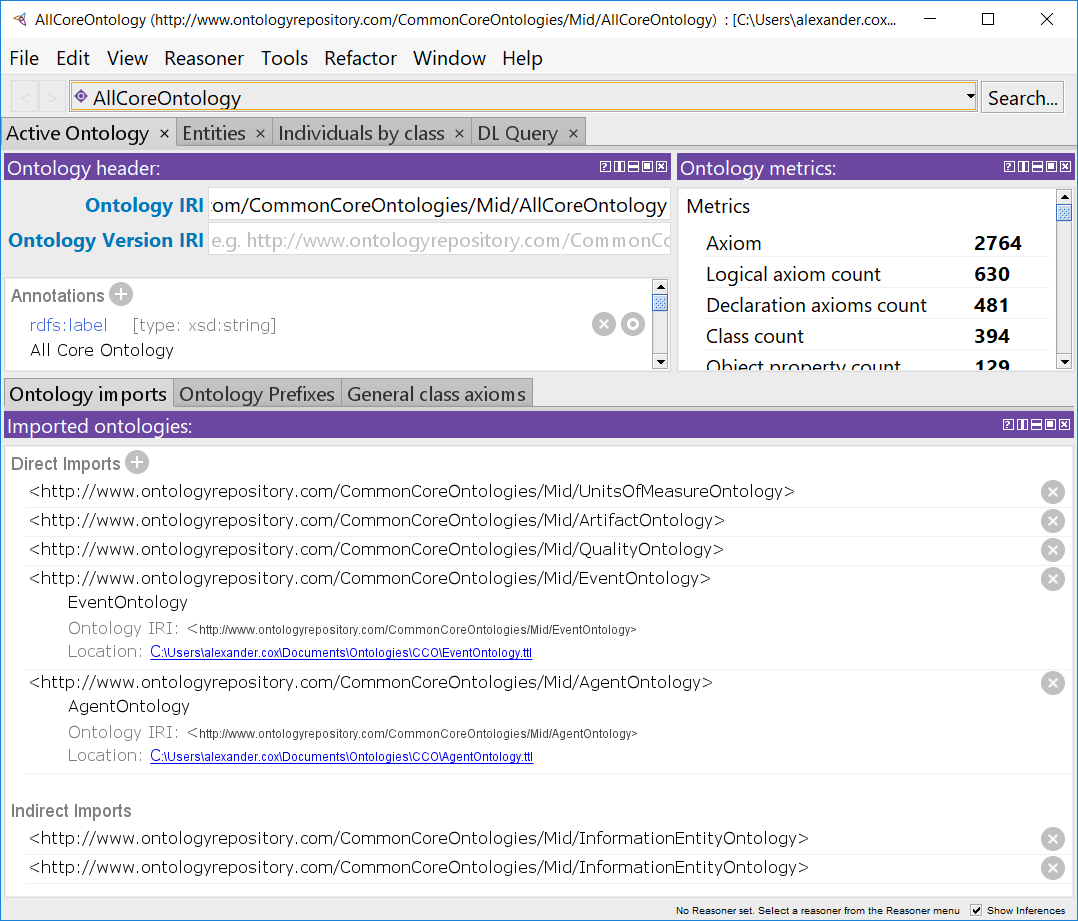


Note that STEP 5 may need to be repeated multiple times the first time you open an ontology that has a lot of imports. Consult Section 2 below for troubleshooting tips.

**STEP 6:** Confirm that the ontology has loaded and that all direct and indirect imports have been properly resolved.

In the image below, only the Event Ontology and the Agent Ontology imports were resolved. The remaining direct and indirect imports were not resolved, as is indicated by the lack of a file location being displayed.

If problems arise during the import resolution process, consult Section 2 below for troubleshooting tips.



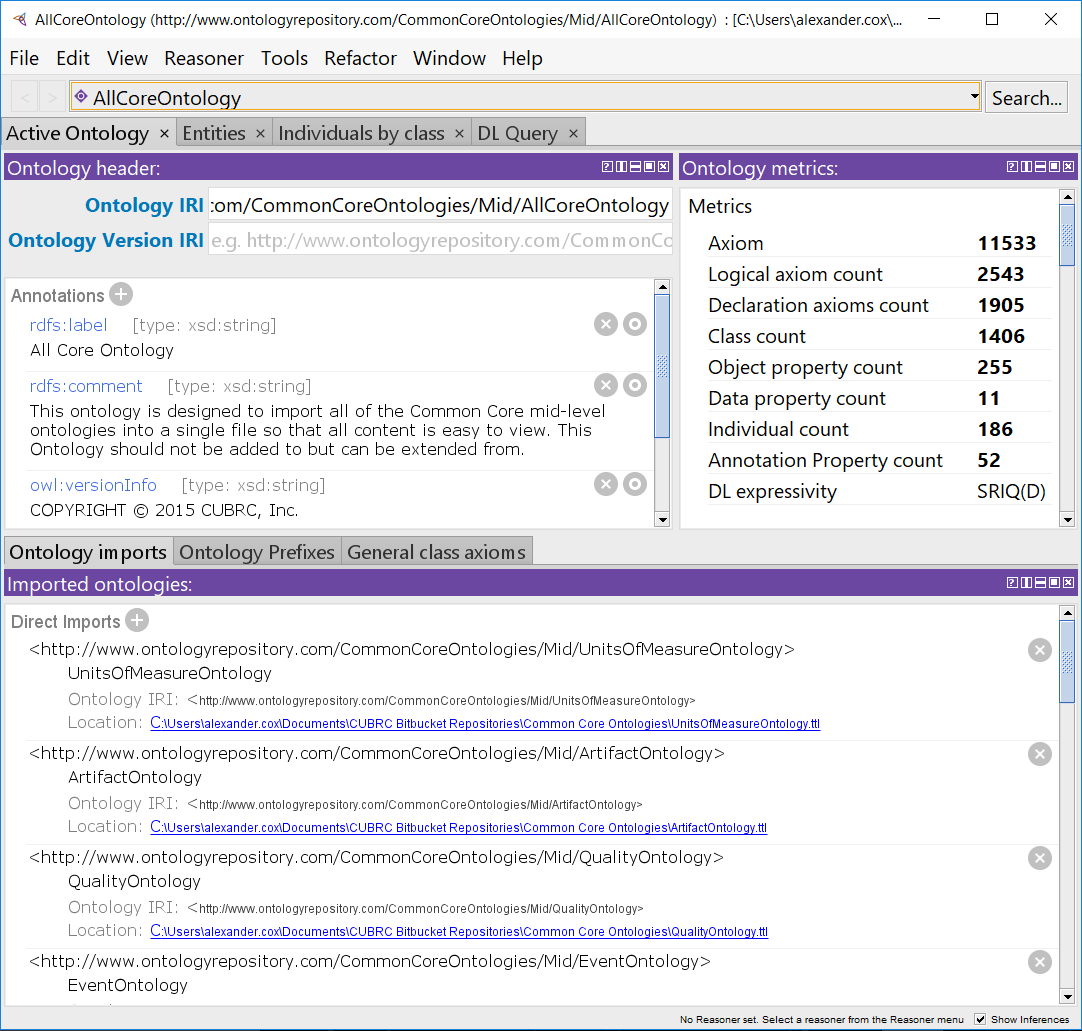
**RESOLVED IMPORTS**

**FAILED IMPORTS**

**FAILED IMPORTS**

**STEP 7:** View the ontology.

You should see a screen similar to the image below. Consult Section 3 of this document for an introduction to browsing ontologies in Protégé.



**Section 2: Loading the Ontologies – Troubleshooting**

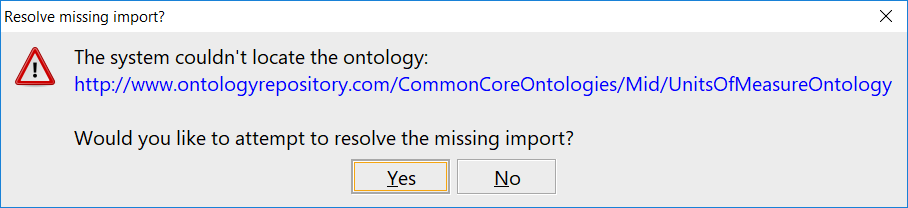
When opening an ontology, Protégé must resolve any imports in order to properly display the active ontology and its dependencies. Protégé will attempt to automatically resolve imports by first consulting the automatically generated catalog file, then searching online for files not located using the catalog, and finally by asking the user to help locate files not found using the first two methods. If this is the first time an ontology has been opened, you will probably need to manually resolve one or more imports. This section provides tips for troubleshooting this process. Once your catalog files are properly configured, Protégé will be able to locate every import file and will load ontologies much faster and without further assistance. Future changes to the catalog file will only need to be made if a file path changes or new ontologies are added.

**TIP #1:** Turn off your internet connection before attempting to load an ontology for the first time.

Protégé will attempt to locate imports by searching online for files that are not found using the catalog. This can become very time-consuming when the ontology URL does not point to an online file. Protégé will continue attempting to resolve the URL until it times out after 20 seconds before asking the user to locate the file manually. This can result in a significant delay the first time loading an ontology with many imports. Turning off your internet connection prior to opening a new ontology significantly speeds up the process by preventing Protégé from waiting for every ontology URL to time out.

With internet access turned off, Protégé should present the “Resolve missing import?” pop-up windows in quick succession which allows the user to resolve imports as quickly as he/she can locate the specified files. Note that Protégé may prompt you to repeat this process the second time you open the same file. Protégé typically creates an empty catalog file the first time, then populates it the second time. Fortunately, with internet access off, it should require less than 5 minutes to complete both iterations of import resolution.

**TIP #1.1:** The first time opening the first ontology in the folder, click ‘No’ to every “Resolve missing import?” request.



Click ‘No’ for all prompts the first time opening an ontology in the folder

This will result in the creation of an empty catalog file. Close the Protégé window that opens after you have denied all import resolution requests, then reopen the same file. Resolve all missing imports during the second attempt. This should result in the successful population of the empty catalog file and no further import resolution should be necessary. You can confirm this by either closing then reopening the ontology another time or by opening the catalog file in a text editor to check its contents.

**TIP #2:** Try and try again.

Protégé may not properly resolve all of the imports on the first, second, or even third attempt. If you have the patience, keep opening the desired ontology, wait for Protégé to prompt you to resolve imports, then check the list of direct and indirect imports to ensure that everything has been located. If not, close Protégé, reopen it, and try opening the desired ontology again. This method will often eventually resolve all of the imports, but can be frustrating and time-consuming. If you attempt this approach, you may want to open the catalog in a text editor in order to ensure that progress is being made during each loading attempt. If progress stalls, keep reading for more troubleshooting tips.

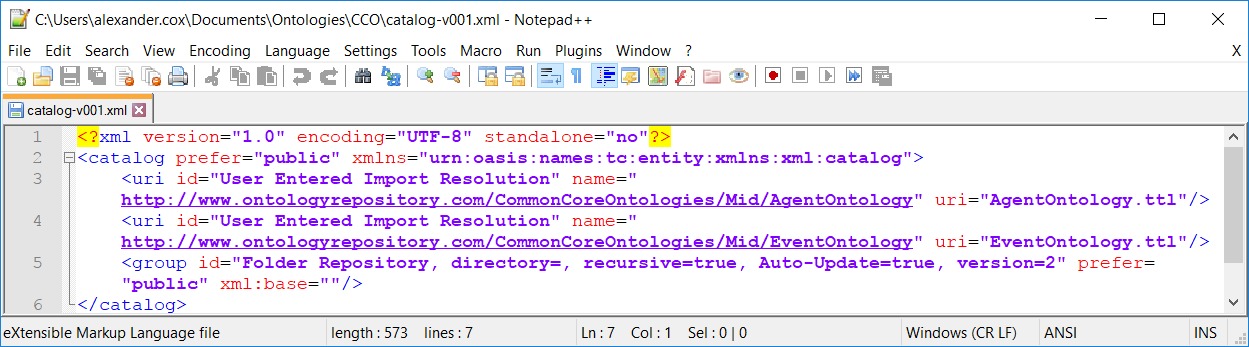
During testing without turning off internet access, it required opening AllCoreOntology.ttl three times and resolving the missing import prompts each time before all of the imports were resolved. This entire process took nearly 30 minutes to complete. Opening AllCoreOntology.ttl on a fourth or subsequent attempt took about 3 seconds, as should be expected.

**TIP #3:** Manually curate the catalog file.

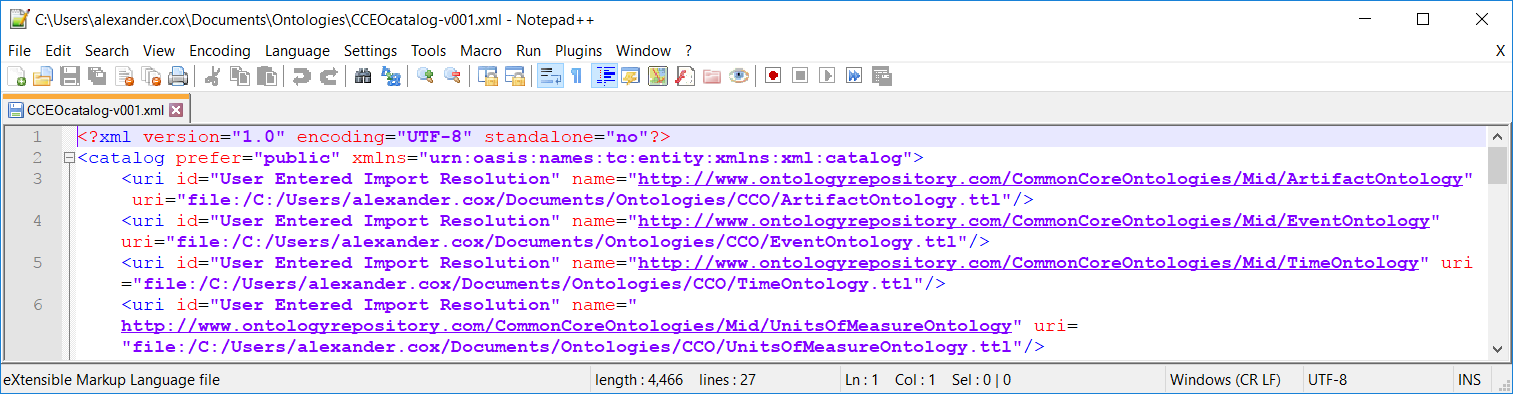
Protégé will automatically generate a catalog file <catalog-v001.xml> in the same folder as the selected ontology file the first time an ontology in that folder is opened. The catalog file is automatically updated when new imports get resolved. If everything works properly, you will never need to look at or think about the catalog; however, if you are reading this, the catalog generation process probably has not worked flawlessly.

*OPTION 1:* Manually add an entry for each ontology to the catalog file.

The format of a given entry will depend on whether the file is located in the same folder as the ontology file you are opening or not. The image below is an example of a partially populated catalog file. The AgentOntology.ttl and EventOntology.ttl files are both located in the same folder as the AllCoreOntology.ttl file that was being opened.



Files located in a different folder will have a file path included as part of their catalog entries. The image below shows a portion of a catalog file for a folder containing only Common Core Extension Ontologies. The ArtifactOntology.ttl, EventOntology.ttl, TimeOntology.ttl, and UnitsOfMeasureOntology.ttl files are all located in a different folder so their uri values specify the full file path instead of just the file name.



OPTION 1 typically requires more effort than OPTION 2, but is an excellent method for resolving persistent import problems.

NOTE: If internet access is on when generating the catalog files, Protégé may load BFO from its online file location instead of from the local file. It is suggested that the local file is used instead of the online one to ensure that no errors will occur if the online file is inaccessible or has been altered. Using OPTION 1 will enable you to change which location BFO is loaded from.

*OPTION 2:* Reuse premade catalog files.

Copy and paste a premade catalog file (such as the ones provided below) into the folder where the relevant ontologies are saved. Next, open the catalog in a text editor and, if necessary, run find and replace searches to change the file paths in the template catalog to match your local file paths. Note that you will only need to do this for files that are located in a different folder, but you will need to do this for each additional folder where imported ontologies are saved.

IMPORTANT: Be sure the name of your catalog file is <catalog-v001.xml> or Protégé will not recognize it. Once your catalog is set up, follow the instructions in Section 1 above to open the desired ontology.

Below are premade catalog files that can be used to quickly create your catalog files using OPTION 2. The first file <CCOGitHub\_catalog-v001.xml> is to be used in a folder containing the Common Core Ontologies and a sub-folder named “imports” containing the upper level ontology files for BFO and RO. This catalog file is designed to work with the ontology file structure on the CCO GitHub page. The second file <CCOSingleFolder\_catalog-v001.xml> is to be used in a folder containing BOTH the Common Core Ontologies and the BFO and RO files without subdirectories. This is the file you should use if you follow TIP #4 below.

**TIP #3.1:** OPTION 2 is very useful if you ever change the location(s) of your ontology folders.

**TIP #4:** Save all ontologies in a single folder.

Protégé creates a catalog file in each folder that a saved ontology is directly opened from. While there are good reasons to use multiple folders to separate groups of ontologies, this can result in a greater upfront effort when opening ontologies for the first time. If you simply want to view the ontologies, storing them in a single folder will simplify the catalog creation and maintenance process.

During testing with internet access off and all ontology files in a single folder, the time required to successfully generate the catalog file from scratch and load the ontologies was reduced to about 5 minutes when an extension ontology that imported more than 20 ontologies. Note that TIP #1 and TIP #2 above applied to the test since it involved generating the catalog file from scratch.

Since all of the ontology files are saved in a single folder, once the imports have been resolved for an ontology that imports all of the other ontologies in the folder, no additional import resolution will be required when opening other ontologies in that folder. Hence, when working with just the CCO, it is recommended to start with AllCoreOntology.ttl. Similarly, when working with more than one CCO extension ontology, identify the file(s) that are not themselves imported by another file to start with.

**TIP #5:** Delete your current catalog file to eliminate some loading errors.

In some rare cases the easiest way to resolve import errors is to delete the existing catalog file and create a new one. In such cases, it is recommended that you save a copy of the current catalog until you are sure that this is the best solution.

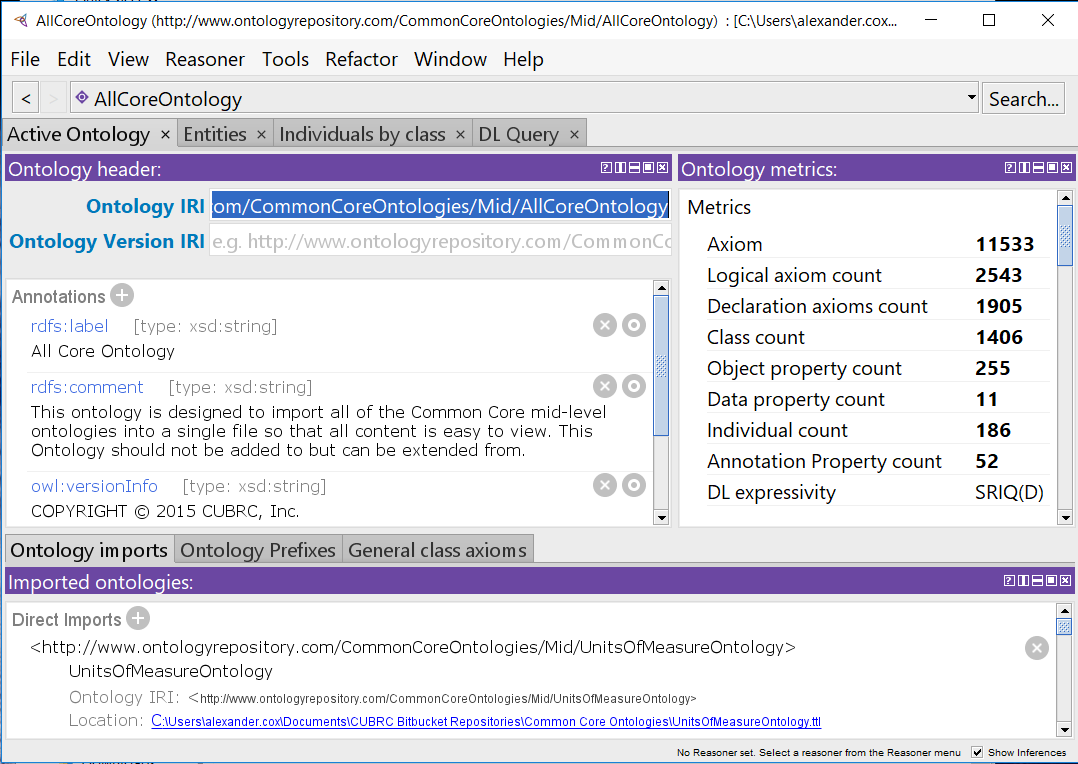
**Section 3: Browsing the Ontologies**

Ontologies are commonly saved as .owl or .ttl files. These can be opened, viewed, and searched using a standard text editor; however, lack of familiarity with the syntax can make viewing the ontologies in a text file challenging. Moreover, the formatting of ontology files makes it difficult to understand the structure of larger or more complex ontologies. This is especially true for ontologies that import and depend on terms from other ontologies since only the terms in the active ontology are stored in the file. Hence, it is recommended that users open the target ontology in an ontology viewer such as the Protégé ontology editor (see Section 1 for instructions). This section introduces users to browsing ontologies using Protégé.

After opening the desired ontology and resolving any imports as per the instructions in the preceding sections, you will see the “Active Ontology” view. This view provides an overview of the selected ontology and looks similar to the image below.

The “Active Ontology” view allows users to see:

1. The name of the active ontology – the ontology you are currently working on
2. Annotations for the active ontology, such as a label, description, and version information
3. Count metrics about the contents of the ontology (NB: these numbers include imported content as well as the active ontology’s content)
4. A list of directly and indirectly imported ontologies and the location of their files



**1**

**2**

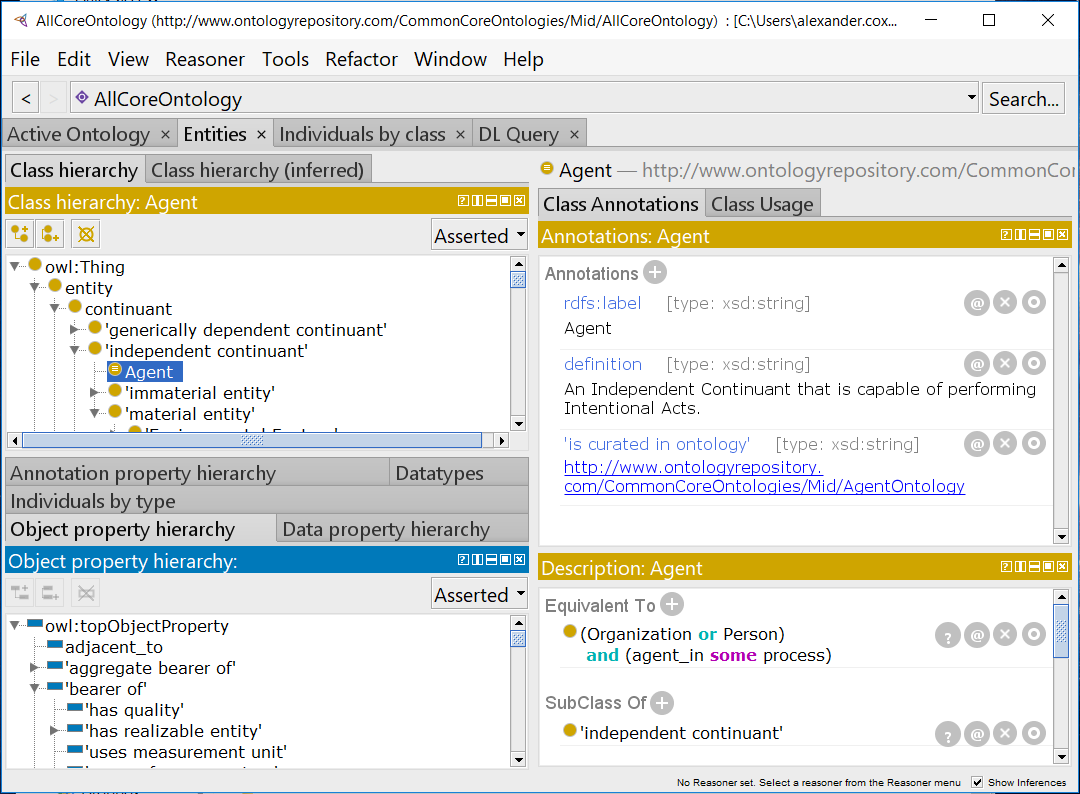
**4**

**3**

To view the contents of the ontology, click on the tab labeled ‘Entities’ in the top left of the window (the red box in the image below) to switch to the “Entities” view. This is the default view for browsing and editing ontologies. It will look similar to the image below.

The “Entities” view allows users to see:

1. The class hierarchy. You can expand or collapse hierarchies by clicking on the triangles to the left of terms or by double-clicking on the terms themselves. Terms that are part of or have been modified by the active ontology are **bolded** to distinguish them from imported terms. (NB: Since the All Core Ontology is designed to aggregate other ontologies and does not contain any terms of its own, nothing will be bolded when it is the active ontology.)
2. Annotations for the currently selected term. Common annotations include labels, definitions, definition sources, comments, and examples of usage. You can also see a list of the selected term’s logical relations to other assertions and terms by clicking on the ‘Class Usage’ tab above this window.
3. Logical axioms that define or restrict the meaning or use of the currently selected term. Note that axioms asserted of a parent class are inherited by its children.
4. The hierarchy of the currently selected property type. Selecting a term here will result in the contents of window 2 and window 3 switching to display the annotations and axioms, respectively, for that term.
5. Individuals and the various property hierarchies. Selecting a different view will change whether individuals or one of the property hierarchies is displayed in window 4.



**5**

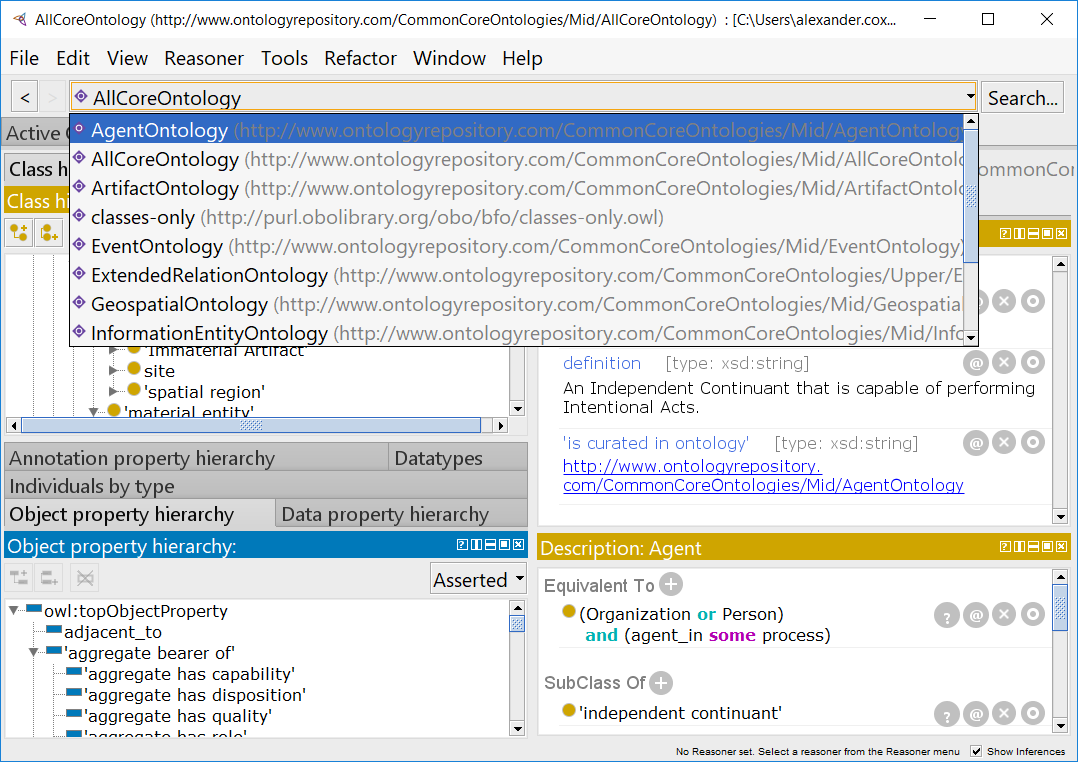
**2**

**3**

**1**

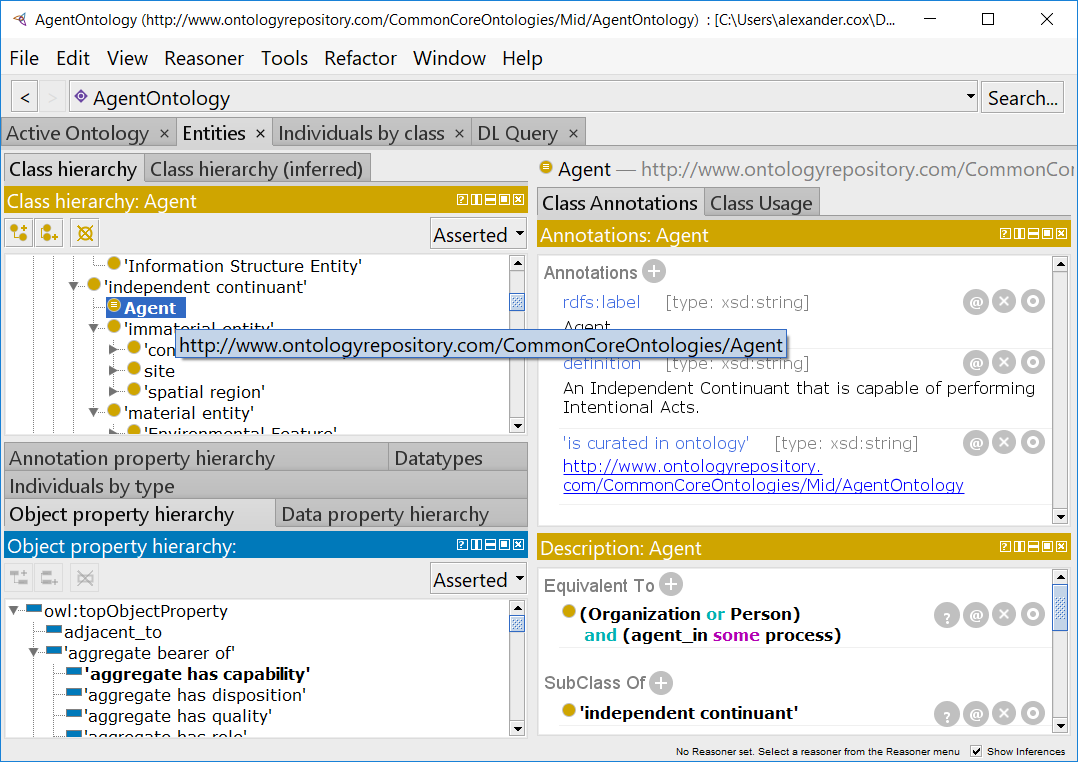
**4**

If you want to view an ontology that is imported by the file you opened, you can do so without opening another Protégé window. Simply click on the name of the active ontology then select the name of the imported ontology you want to view. This ontology will then become the active ontology until you switch again. Switching which ontology is active is illustrated in the following image:



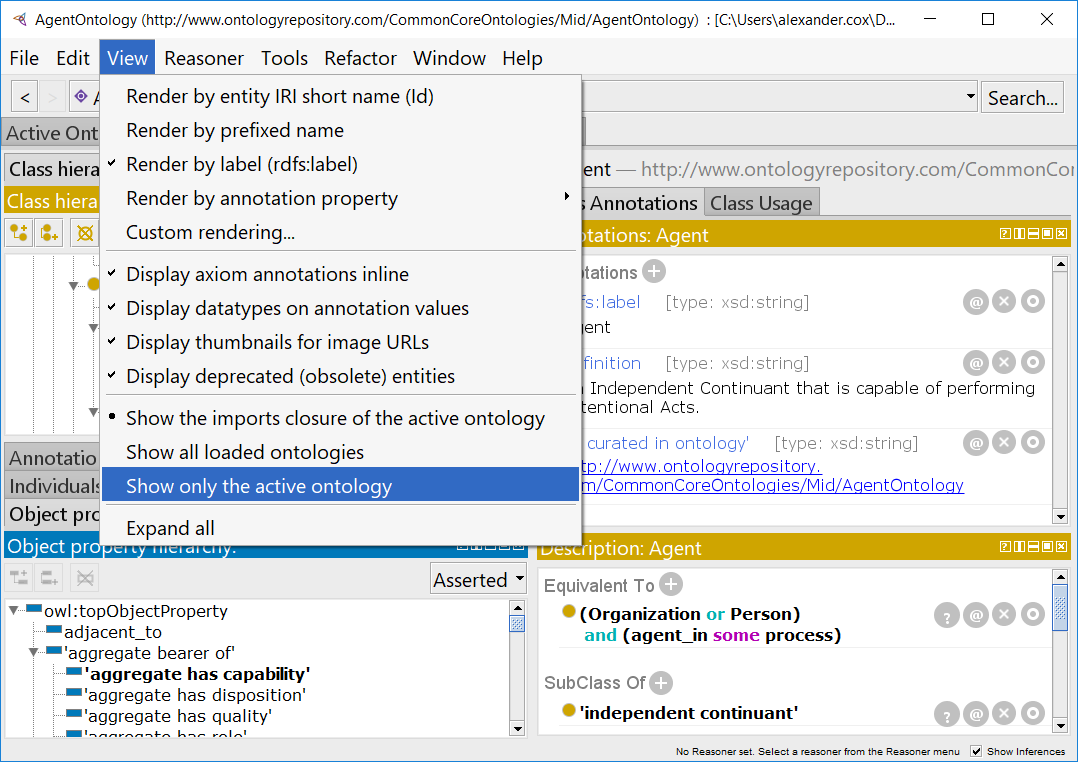
In the image below, the Agent Ontology is now the active ontology after having been selected as shown in the image above. As a result, terms and axioms asserted in the Agent Ontology are now **bolded**.

Note that additional information, such as the full IRI of a term or which ontology an annotation or axiom is asserted in, can be viewed by holding the cursor over a term, annotation, or axiom. The full IRI of ‘Agent’ is displayed in the image below.



If you want to view only the terms in the active ontology with a bare minimum of imported content, click on ‘View’ then ‘Show only the active ontology’ as shown in the image below. This can make it significantly easier to find and review ontology content, especially when it is further from the root node or is not all concentrated under a single parent class.

If you want to quickly expand the term hierarchies without clicking your mouse to death, click on ‘View’ then ‘Expand All’. The term hierarchies will be fully exploded. This can be useful for exploring the ontology or quickly locating bolded terms from the active ontology.



If terms are not displayed in your preferred format, click on ‘View’ then select one of the ‘Render by…’ options. This will change how terms are displayed. It is generally easiest to view terms using their labels, which is the option checked in the image above.

If you are looking for a specific term, use the search feature by clicking on ‘Search…’ in the upper right corner or by using the keyboard shortcut Ctrl+F to launch a search window. Results are grouped based on the part of the ontology the match is found in. Priority is given to results for the displayed name. See the image below for an example search result. If you do not see the result you are looking for, check the box labeled ‘Show all results’ to make sure it is not further down the list of results. Once you find a result you want, double-click on it to be automatically taken to the relevant entity. (Pro Tip: if you use the arrow keys to select your desired result then hit ‘Enter’, the search box will automatically close and you will be taken to your selection.)

