c-Mantic: a [Cytoscape](#) [Semantic Web](#) Plugin

c-Mantic: User Manual

Introduction - What is the c-Mantic Plugin?

c-Mantic ("seMantic") is a [Cytoscape](#) plug-in that uses [Semantic Web](#) technologies to support the analysis of linked data networks.

The c-Mantic plug-in is useful for analyzing OWL/RDF data as Cytoscape networks. The OWL/RDF data can come from remote knowledgebases (SPARQL endpoints), imported from OWL/RDF data files, or from transforming the data in any Cytoscape network.

The work is part of the Virtual Liver (v-Liver™) project at the National Center for Computational Toxicology in the US Environmental Protection Agency (EPA). Visit http://www.epa.gov/ncct/v-liver/ for more information. The c-Mantic plugin was implemented by Lockheed Martin under the guidance of the US EPA.

Key Features

The c-Mantic plug-in can be used to,

- Visualize OWL/RDF data as Cytoscape networks
- Query data from remote SPARQL endpoints and local knowledgebases
- Import OWL/RDF files (N3 or OWL/RDF/XML format)
- Export Cytoscape networks as OWL/RDF files
- Create local knowledge bases from Cytoscape networks
- Apply inference rules to local knowledge bases

Installation
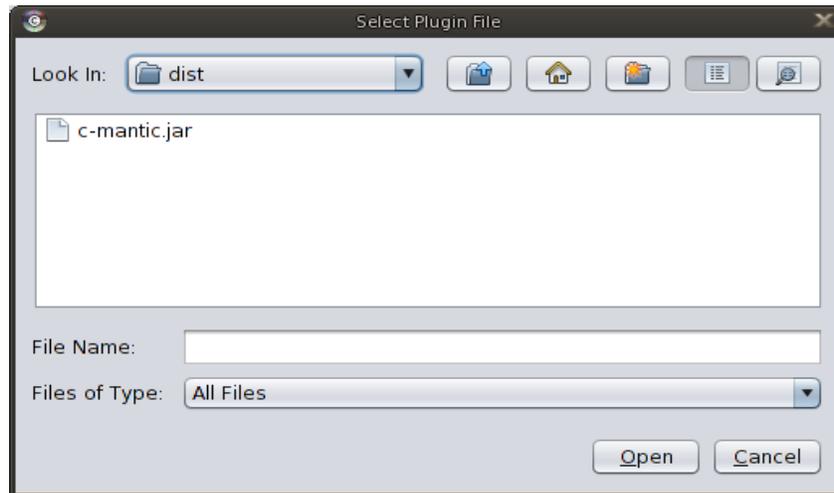
7

1. Download and install Cytoscape version 2.8.* or later
2. Download the plugin (c-mantic.jar) and save it a folder.
3. Go to the Cytoscape Plugins Menu and select "Install Plugin from File"
4. Open the folder and select the plugin file
5. The c-Mantic icon will now appear under the "Plugins" menu

Accessing c-Mantic Features

Features of c-Mantic are accessed through individual tabs in the main plugin window.
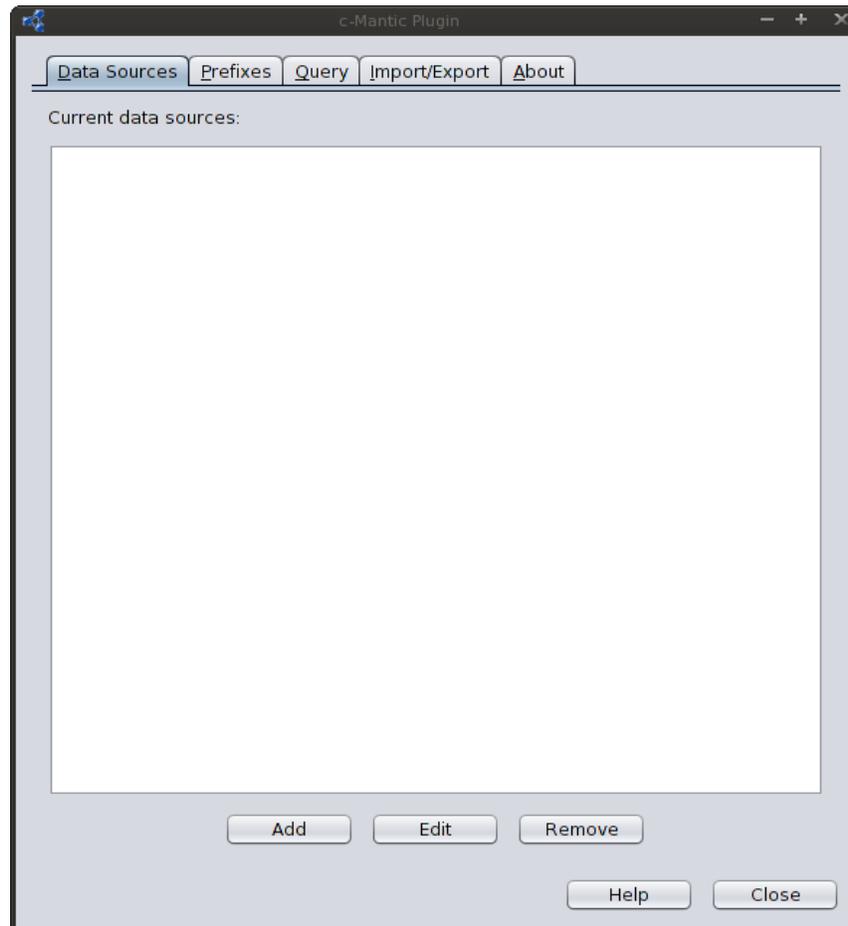

Figure 1. c-Mantic Plugin

These features include defining data sources and prefixes that are used in SPARQL queries. Other features include the ability to import, export and translate OWL/RDF data. Details of each tab are described below.

Data Sources Tab

A Data Source defines properties of an OWL/RDF knowledge base to the plugin. The plugin can use a data source as part of a query, import or export function. c-Mantic maintains a list of previously defined data sources under the Data Sources tab. Properties of a data source include its name, type, and location.

The Data Source Editor lets one create a new data source or modify an existing data source.

To create a new data source,

1. Click the Data Source tab.
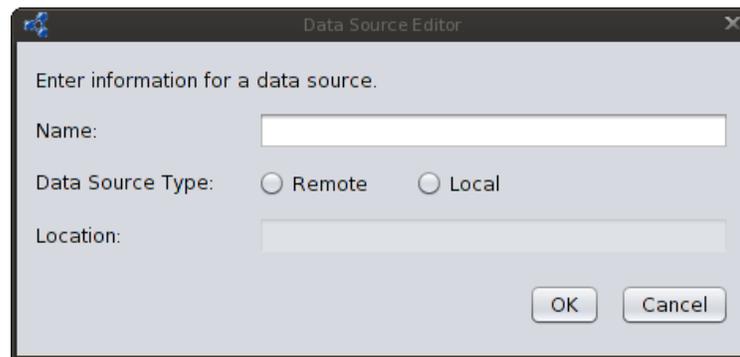2. Click the Add button to open the Data Source Editor.

Figure 2. Add - Data Source Editor

3. Enter a data source name. The name can be a long, descriptive name but **must be unique** among data sources. The name will be used to refer to the data source in other c-Mantic functions such as queries.
4. Select the data source type. There are two types of data source, remote and local.
   a. A remote data source refers to a SPARQL endpoint, a knowledge base that can accept SPARQL queries.
   b. A local data source is a folder on your local file system. You can import OWL/RDF files or export Cytoscape networks into a local data source.
5. Enter the location for the data source. The location depends on the type of data source and **must also be unique** among data sources.
   a. The location for a remote data source looks like a web site adress. Be sure to include the "http://" in the location text field.
   b. The location for a local data source is a folder created on your local file system. Enter a simple word or phrase for the folder name.
6. Click the OK button to create and save the data source. The data source name will be added to the list of data sources under the Data Sources tab.

To edit a data source,

1. Select a name from the list of data sources under the Data Source tab.
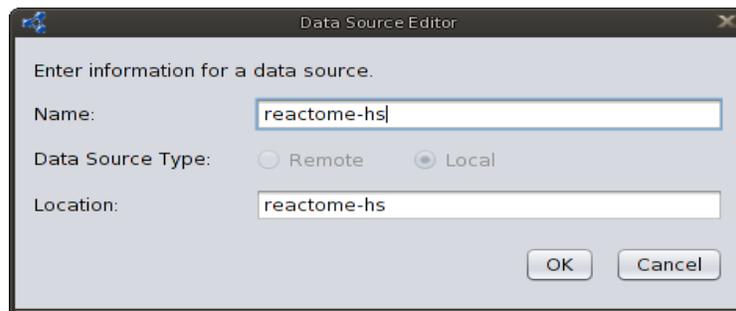2. Click the Edit button to open the Data Source Editor.

Figure 3. Edit - Data Source Editor

Edit the data source to change its name or location. Remember, the name and location must be unique. Note, you cannot change the data source type. Click the OK button to save your changes.

To remove a data source from the list,

1. Select a name from the list of data sources under the Data Sources tab.
2. Click the Remove button and confirm the name of the data source to be removed.

Prefixes Tab

c-Mantic maintains a list of namespace prefixes under the Prefixes tab. Prefixes are abbreviations for namespace names in OWL/RDF data and SPARQL queries. c-Mantic includes several predefined namespace prefixes that are commonly used in the OWL/RDF community.
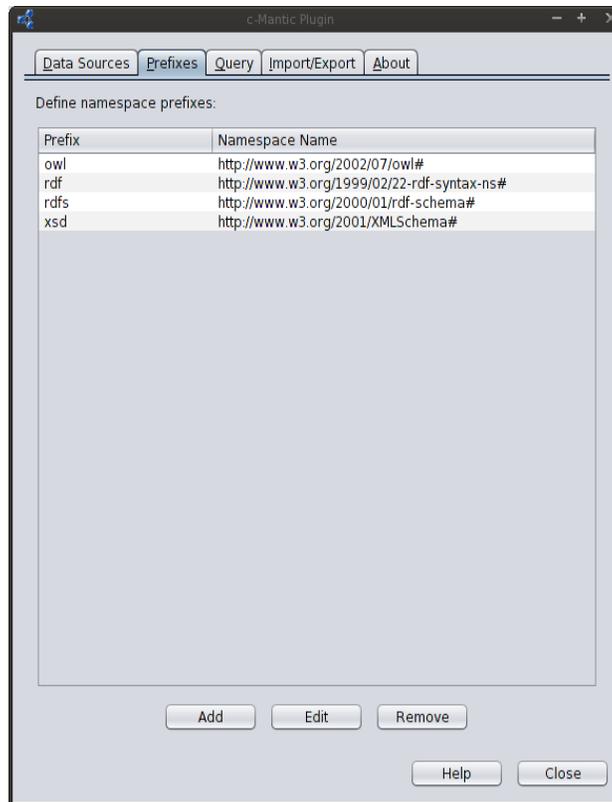

Figure 4. Prefixes Tab

To create a new prefix,

1. Click the Prefixes tab.
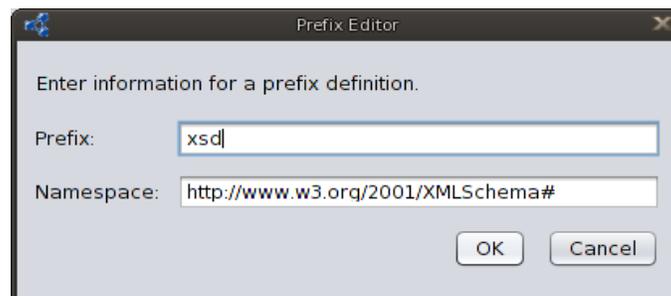2. Click the Add button to open the Prefixes Editor.

Figure 5. Prefix Editor

The Prefix Editor lets one create or modify a namespace prefix.

1. Enter a namespace prefix. Prefixes are often short combinations of letters and numbers and must be unique.

2. Enter a namespace name for an OWL/RDF vocabulary. Click the OK button to save the namespace prefix. The prefix will be added to the list under the Prefixes tab.

To edit a namespace prefix,

1. Select a prefix from the list of namespace prefixes under the Prefixes tab.

2. Click the Edit button to open the Prefix Editor.

3. Edit the prefix or the namespace name.

4. Click the OK button to save the namespace prefix.

To remove a prefix from the list,

1. Select a prefix to be removed by clicking a row of a prefix from the list under the Prefixes tab.

2. Click the Remove button and confirm the prefix to be removed.

Query Tab

The Query Tab is where you enter a query and specify the data source (or knowledgebase) to query. You may use prefixes to make your query and the results easier to read.

Select a Data Source from the drop-down list to specify where c-Mantic should submit your query. You can create a query in the Editor panel.

Editor Panel

Use the Editor panel to create a new query or retrieve a saved query. The Recent Queries list holds the ten most recently used queries. You can type a new query in the Editor panel text area. The Save button allows you to save the query to a file. The Open button allows you to retrieve a saved query. The Delete button allows you to delete a query file from the system.

The Execute button submits the query in the text area to the selected Data Source. You can stop a running query by clicking the Cancel button.

If you receive a "Connection timed out" error, you may need to configure Cytoscape's proxy server settings in order to query remote data stores. A proxy server is a network service that administrators can use to maintain network security and performance. You can configure Cytoscape to use your network's proxy server from the Cytoscape Edit menu (**Edit → Preferences → Proxy Server**). The Cytoscape manual has instructions for configuring the proxy server settings. Ask your network administrator for the host name and port number of your HTTP proxy server.

Figure 6. Query Tab
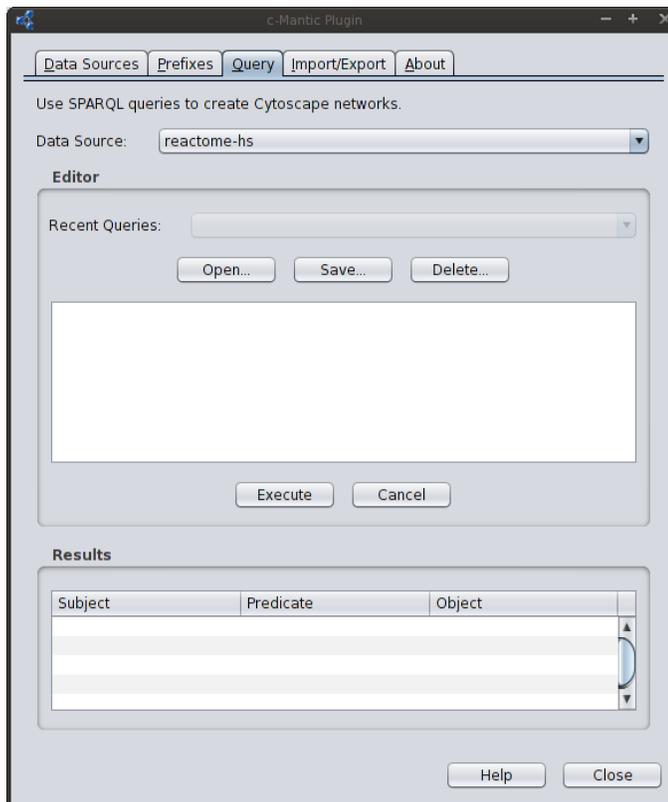
Results Panel

Data returned from a Data Source appears in the Results panel. The type of query submitted to the Data Source determines the type of data returned.

SPARQL Select queries return tuples. The format depends on the variables in the SELECT clause and the graph pattern in the WHERE clause.

SPARQL Construct and Describe queries return graphs containing individual OWL/RDF statements. The OWL/RDF subject, predicate, and object of each graph statement will appear in the rows of the Results Panel. c-Mantic also converts the graph into a Cytoscape network.

Results from a SPARQL Ask query appear in a pop-up window.

Note:

For more information on SPARQL queries, see the web site http://www.w3.org/TR/rdf-sparql-query/.

Import/Export Tab

Import Panel

The Import Panel allows you to load an OWL/RDF file (N3 or OWL/RDF/XML format) to a local data source.

To import an OWL/RDF file to a local data source,

1. Select a local data source.
2. Click the Import button.
   a. If a local data source is not selected, you will be asked to create a local data source.
   b. If you choose to create a new local data source, the Data Source Editor will open. Enter the name and location for the local data source and click the OK button.
3. A file chooser will open. Select the OWL/RDF file you wish to import and then click the Import button in the file chooser.
4. A dialog will confirm that the file was successfully imported. Click the OK button.

Figure 7. Import/Export Tab

Export Panel

The Export Panel allows you to save a Cytoscape network to either an OWL/RDF file (N3 or OWL/RDF/XML format) or a local data source.

To export a Cytoscape network to an OWL/RDF file,

1. Select a Cytoscape network.
2. Click the "OWL/RDF File" button for the type of Export.
3. Click the Export button.
4. In the Export file chooser, confirm that the proper OWL/RDF file type is selected in the file type box.
    a. Select ".n3" for an N3 file.
    b. Select ".rdf" for an OWL/RDF/XML file.
5. You may either create a new OWL/RDF file or overwrite a previous file.
    a. To create a new OWL/RDF file, enter a unique name in the File name text box.
    b. To overwrite a previous OWL/RDF file, select the file.
6. Click the Export button in the file chooser.
7. An input box may appear to ask you for a Base URI to complete the export. Enter a valid URI and hit the OK button.
8. A dialog will confirm that the Cytoscape network was successfully exported to the OWL/RDF file. Click the OK button.

To export a Cytoscape network to a local data source,

1. Select a Cytoscape network.
2. Click the "Local Data Source" button for the type of Export.
3. Select a local data source.
4. Click the Export button.
    a. If a local data source was not selected, you will be asked to create a local data source in order to export a network to the newly created data source.
    b. If you choose to create a new local data source, the Data Source Editor will be opened. Enter the name and location for the local data source and click the OK button.
5. An input box may appear to ask you for a Base URI to complete the export. Enter a valid URI and hit the OK button.
6. A dialog will confirm that the Cytoscape network was successfully exported to the local data source. Click the OK button.

# c-Mantic Tutorial

Using c-Mantic for Reactome Pathways

This tutorial provides steps by step instructions on using c-Mantic to analyze Reactome pathways. Reactome is an extensive resource of biological pathways which is available in OWL format.

Creating the KB

1. If you have not installed the c-Mantic plugin then please follow the instructions provided here. Run Cytoscape and start the c-Mantic plugin
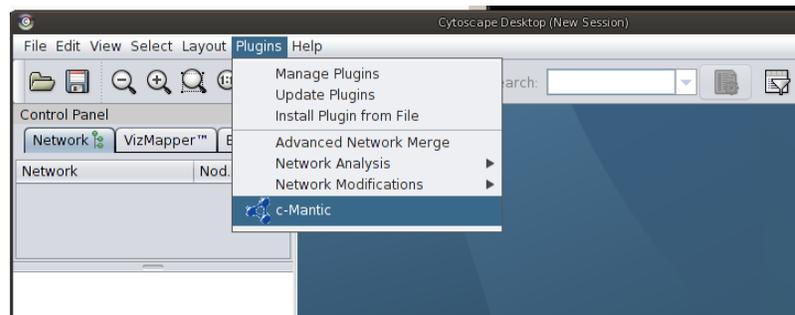

2. Select the "Data Source" tab in the main c-Mantic window and "Add" a KB for Reactome pathways – call it "reactome_hs"
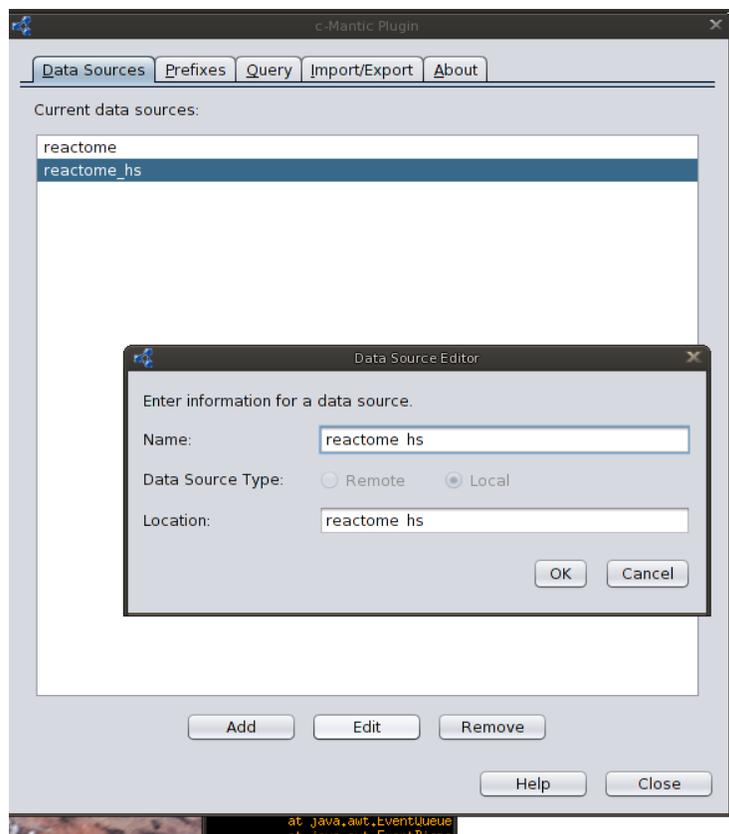
3. Download the Reactome human pathways provided in BioPAX Level 3 format from here (reactome-homo-sapience.owl). Select the "Import/Export" tab in c-Mantic and select "reactome_hs" as the "Local Data Source." Click the "Import" button and select the Reactome file (reactome-homo-sapience.owl). This may take a few minutes depending on the speed of your computer – when it completes c-Mantic will let you know.
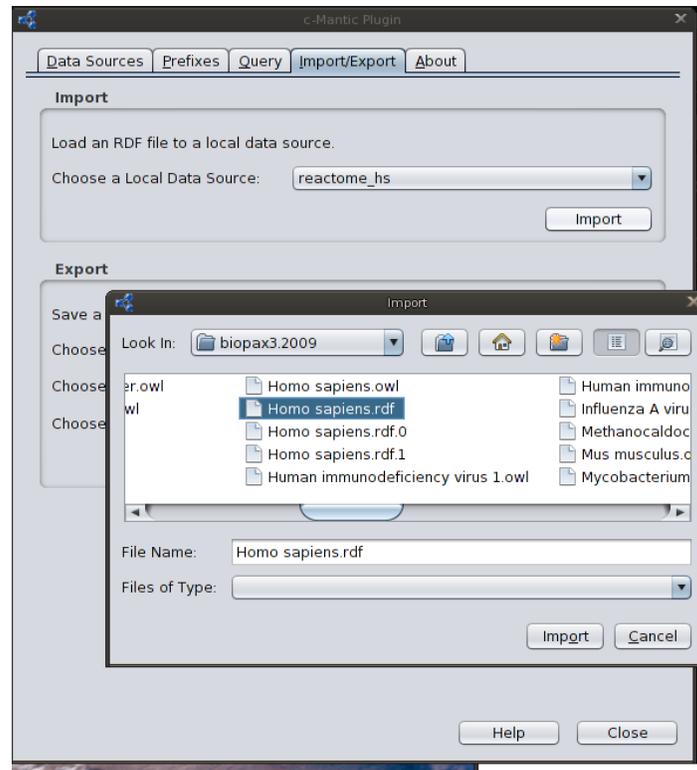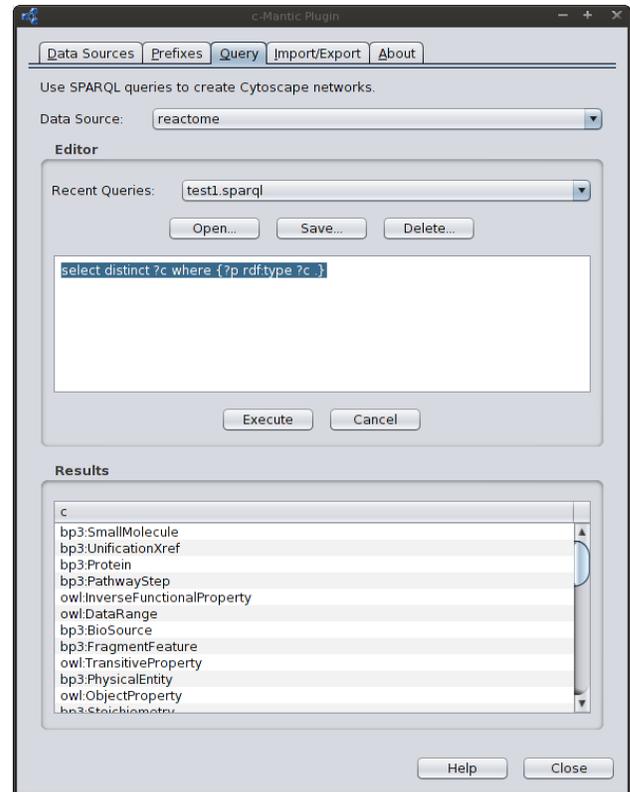

4. Select the "Prefixes" tab and add "bp3" as an abbreviation for: http://www.biopax.org/release/biopax-level3.owl#
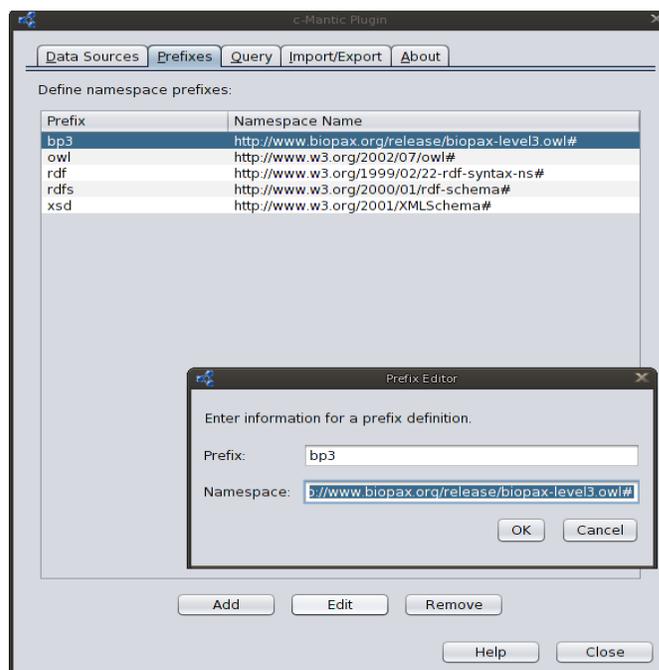
5. Now you can query the Reactome KB in the "Query" tab. Choose "reactome_hs" as the "Data Source" and enter the following SPARQL in the input area:

    ```
    select distinct ?c where {?p
    rdf:type ?c .}
    ```

6. Before visualizing Cytoscape networks, download this vizmap properties file – which has been customized for viewing BioPAX Level 3 information using c-Mantic. Load this file from the Cytoscape menu: "File -> Import -> Vizmap Property File"
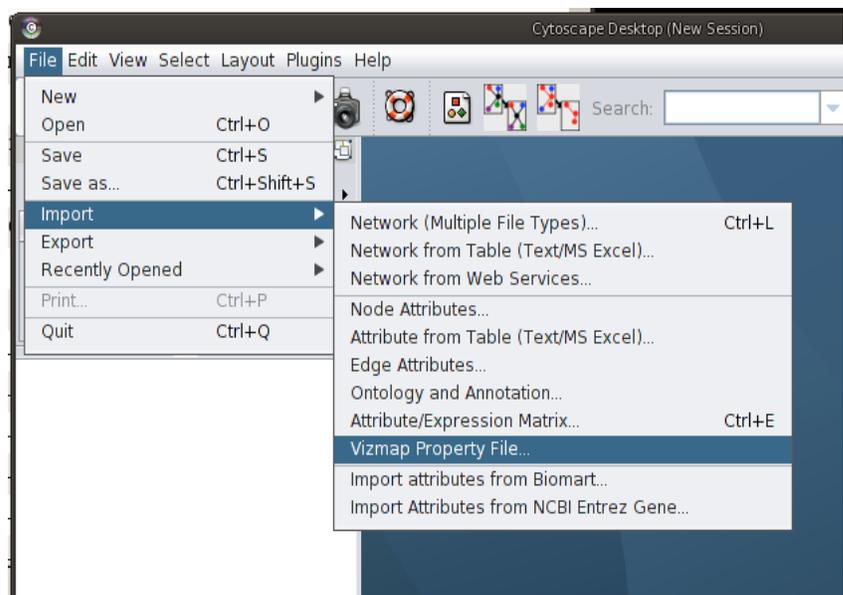
7. Select the "Query" tab in c-Mantic, choose data source "reactome_hs". Enter the following SPARQL in the query input area to visualize components of the cell cycle pathway. (For a newer version of Reactome, you will need the URI for the cell cycle pathway.)

```
construct {
 ?pax rdf:type ?pt; bp3:name ?pn .
 ?pax bp3:pathwayComponent ?pc1; bp3:pathwayComponent ?pc2 .
 ?pc1 rdf:type ?pc1t; bp3:name ?pc1n .
 ?pc2 rdf:type ?pc2t; bp3:name ?pc2n .
 ?pax bp3:pathwayOrder ?ps1; bp3:pathwayOrder ?ps2 .
 ?ps1 rdf:type ?ps1t .  ?ps2 rdf:type ?ps2t .
 ?ps1 bp3:stepProcess ?pc1; bp3:nextStep ?ps2 .
 ?ps2 bp3:stepProcess ?pc2 .
} where {
 ?pax rdf:type ?pt; bp3:name ?pn .
 ?pax bp3:pathwayComponent ?pc1; bp3:pathwayComponent ?pc2 .
 ?pc1 rdf:type ?pc1t; bp3:name ?pc1n .
 ?pc2 rdf:type ?pc2t; bp3:name ?pc2n .
 ?pax bp3:pathwayOrder ?ps1; bp3:pathwayOrder ?ps2 .
 ?ps1 rdf:type ?ps1t .  ?ps2 rdf:type ?ps2t .
 ?ps1 bp3:stepProcess ?pc1; bp3:nextStep ?ps2 .
 ?ps2 bp3:stepProcess ?pc2 .
filter(?pax = <http://www.reactome.org/biopax#Cell_Cycle__Mitotic>)
}
```

When c-Mantic completes this query it will show the number of results and will ask if you want to visualize these as a Cytoscape network. If you select "yes" then these results will appear as a new Cytoscape network. (This is helpful when a query produces millions of results.) You can customize this network using our settings by choosing the "BP3" visual style (additional information is available on the Cytoscape website).

8. For a more intuitive view of cell cycle progression try the following query.

```
construct {
 ?pc1 rdf:type ?pc1t; bp3:name ?pc1n .
 ?pc2 rdf:type ?pc2t; bp3:name ?pc2n .
 ?pc1 bp3:nextStep ?pc2 .
} where {
 ?pax rdf:type ?pt; bp3:name ?pn .
 ?pax bp3:pathwayComponent ?pc1; bp3:pathwayComponent ?pc2 .
 ?pc1 rdf:type ?pc1t; bp3:name ?pc1n .
 ?pc2 rdf:type ?pc2t; bp3:name ?pc2n .
 ?pax bp3:pathwayOrder ?ps1; bp3:pathwayOrder ?ps2 .
 ?ps1 rdf:type ?ps1t .  ?ps2 rdf:type ?ps2t .
 ?ps1 bp3:stepProcess ?pc1; bp3:nextStep ?ps2 .
 ?ps2 bp3:stepProcess ?pc2 .
filter(?pax = <http://www.reactome.org/biopax#Cell_Cycle__Mitotic>)
}
```

Here we have used SPARQL to construct a network in which pathway components are ordered using the "nextStep" relationship.
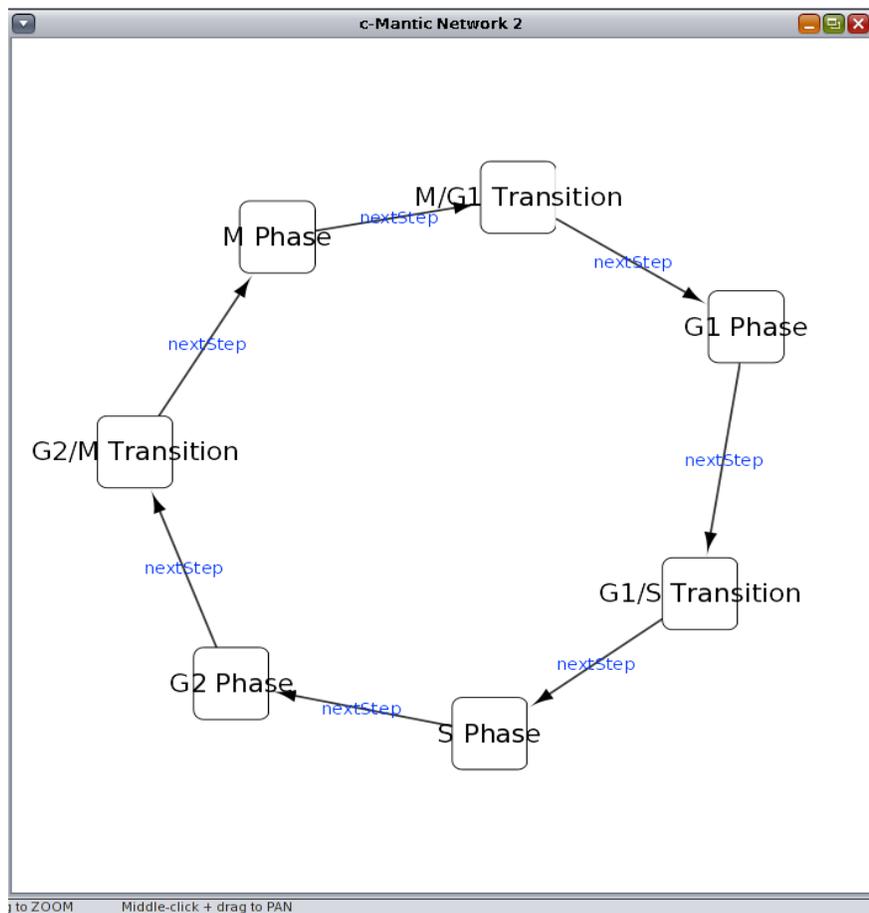
9.   To visualize additional components of cell cycle stages try the following query:

```
construct {
 ?pc1 rdf:type ?pc1t; bp3:name ?pc1n .
 ?pc2 rdf:type ?pc2t; bp3:name ?pc2n .
 ?pc1 bp3:nextStep ?pc2 .
 ?pc1 bp3:pathwayComponent ?pc1c .
 ?pc1c rdf:type ?pc1ct; bp3:name ?pc1cn .
 ?pc2 bp3:pathwayComponent ?pc2c .
 ?pc2c rdf:type ?pc2ct; bp3:name ?pc2cn .
}
where {
 ?pax rdf:type ?pt; bp3:name ?pn .
 ?pax bp3:pathwayComponent ?pc1; bp3:pathwayComponent ?pc2 .
 ?pc1 rdf:type ?pc1t; bp3:name ?pc1n .
 ?pc2 rdf:type ?pc2t; bp3:name ?pc2n .
 ?pax bp3:pathwayOrder ?ps1; bp3:pathwayOrder ?ps2 .
 ?ps1 rdf:type ?ps1t .   ?ps2 rdf:type ?ps2t .
 ?ps1 bp3:stepProcess ?pc1; bp3:nextStep ?ps2 .
 ?ps2 bp3:stepProcess ?pc2 .
 ?pc1 bp3:pathwayComponent ?pc1c .
 ?pc1c rdf:type ?pc1ct; bp3:name ?pc1cn .
 ?pc2 bp3:pathwayComponent ?pc2c .
 ?pc2c rdf:type ?pc2ct; bp3:name ?pc2cn .
 filter(?pax = <http://www.reactome.org/biopax#Cell_Cycle__Mitotic>)
}
```

10.   Finally, the Cyclin E associated biochemical interactions during G1/S transition are visualized using this query:

```
construct {
 ?pc1 bp3:left ?pc1l; bp3:right ?pc1r .
 ?pc1l rdf:type ?pc1lt; bp3:name ?pc1ln .
 ?pc1r rdf:type ?pc1rt; bp3:name ?pc1rn .
}
where {
 ?pax rdf:type ?pt; bp3:name ?pn .
 ?pax bp3:pathwayComponent ?pc1; bp3:pathwayComponent ?pc2 .
 ?pc1 rdf:type ?pc1t; bp3:name ?pc1n .
 ?pc1 bp3:left ?pc1l; bp3:right ?pc1r .
 ?pc1l rdf:type ?pc1lt; bp3:name ?pc1ln .
 ?pc1r rdf:type ?pc1rt; bp3:name ?pc1rn .
filter(?pax=<http://www.reactome.org/biopax#
 Cyclin_E_associated_events_during_G1_S_transition_>)
}
```

Appendix A: Acronyms and Support

Acroynms

Acronyms

OWL Web Ontology Language

OWL/RDF Resource Description Framework

OWL/RDFS Resource Description Framework Schema

URI Uniform Resource Identifier

SPARQL SPARQL Protocol And Resource Query Language (Pronounced "sparkle.")

Supporting Documentation

Visit the following web sites for additional help and information.

Table 1.

Java	http://java.com/en/download/help/index.html
Cytoscape	http://www.cytoscape.org/
OWL/RDF	http://www.w3.org/TR/rdf-primer/
N3	http://www.w3.org/2000/10/swap/Primer.html
SPARQL	http://www.w3.org/TR/rdf-sparql-query/

Appendix B: Third-party Licenses

The c-Mantic plug-in includes third-party open source software components. The copyright and license for each is shown below.

JavaHelp

https://javahelp.dev.java.net/

JavaHelp License

The GNU General Public License (GPL) Version 2, June 1991

Sesame

http://www/openrdf.org/

SwingWorker

http://swingworker.dev.java.net

SwingWorker License

Copyright (c) 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved. Use is subject to license terms below. Sun, Sun Microsystems and the Sun logo are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Notice: This product is covered by U.S. export control laws and may be subject to the export or import laws in other countries. These laws may restrict the fields of use for this software and may require you to secure government authorization.

GNU LESSER GENERAL PUBLIC LICENSE. Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]