2201 Broadway, Suite 715
Oakland, CA 94615
Phone: 510-452-2000
www.franz.com

Using TopBraid Composer with AllegroGraph

Using TopBraid Composer with AllegroGraph

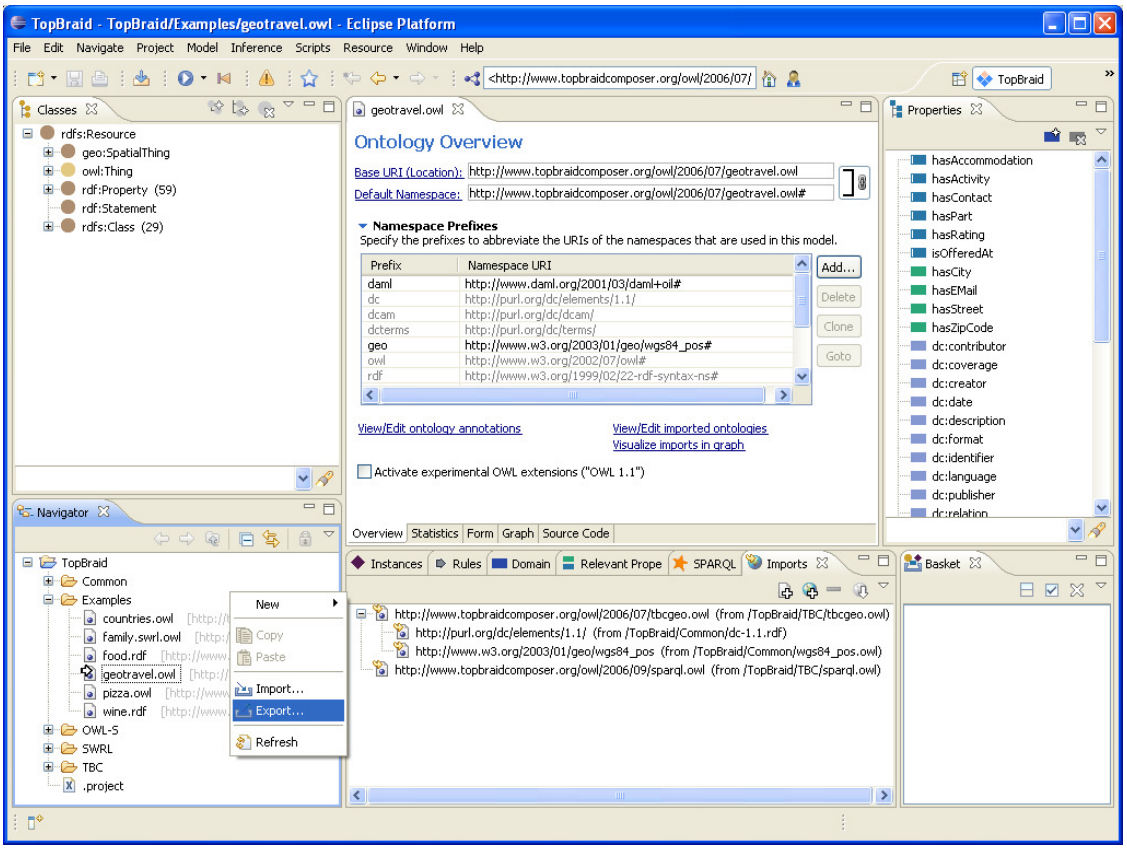
Getting Started with AllegroGraph Java Edition

TopBraid Composer communicates with AllegroGraphServer, so you will need to install both products and start them independently. Please ensure that you have the latest versions of TBC (http://www.topquadrant.com/products/TB_Composer.html) and AllegroGraph (http://www.franz.com/downloads/clp/ag_survey). It is also recommended that you periodically check both for software updates (see http://www.franz.com/agraph/support/documentation/current/server-installation.html#updater).

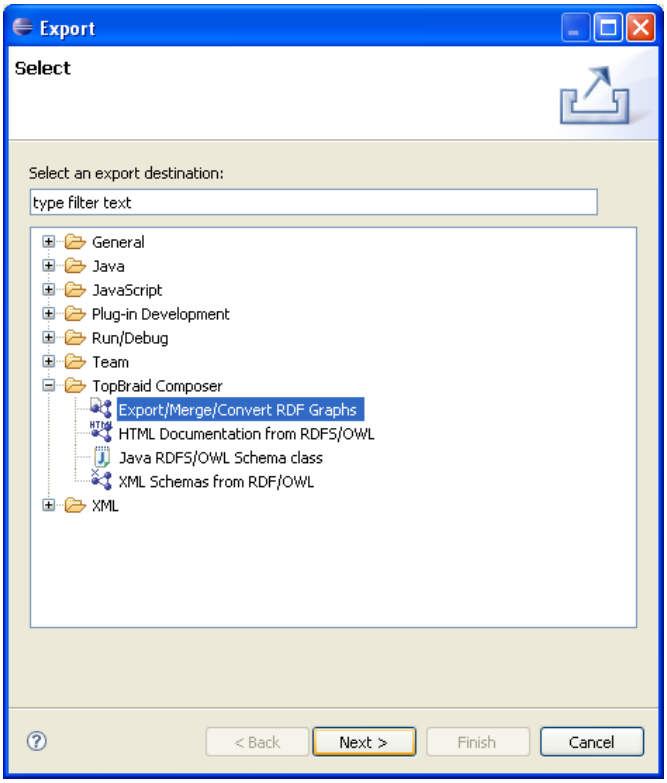
Start the AllegroGraphServer running on some host and port (you'll need to know these later when connecting from TBC – the defaults are localhost and port 4567). For details see http://www.franz.com/agraph/support/learning/Starting-a-server-manually.lhtml.

Exporting an ontology to an AllegroGraph database

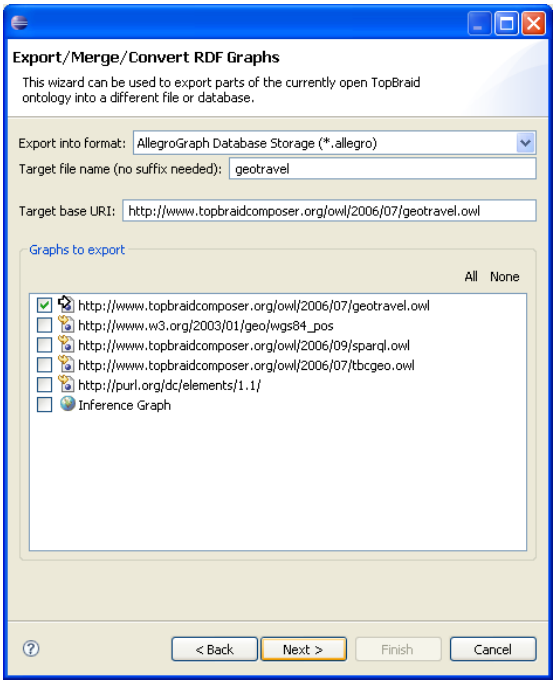
Start TBC and Open an existing OWL file (e.g., geotravel.owl in your workspace\TopBraid\Examples directory), then right click in the Navigator pane and select "Export..."



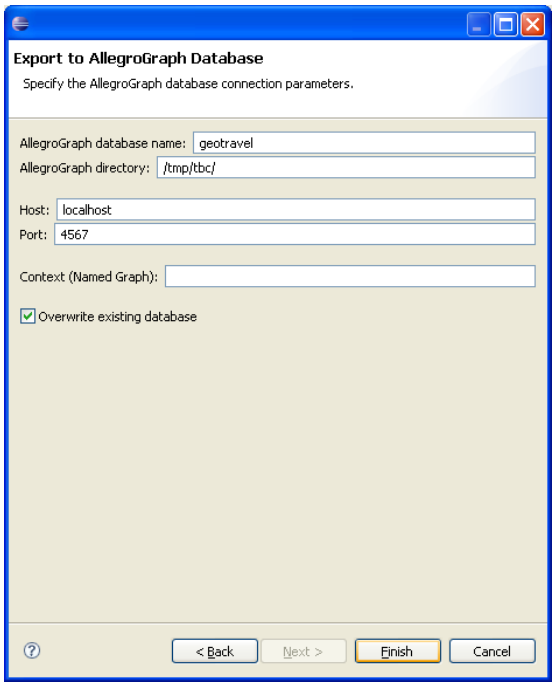
Then select "Export/Merge/Convert" under the TopBraid Composer export destination:



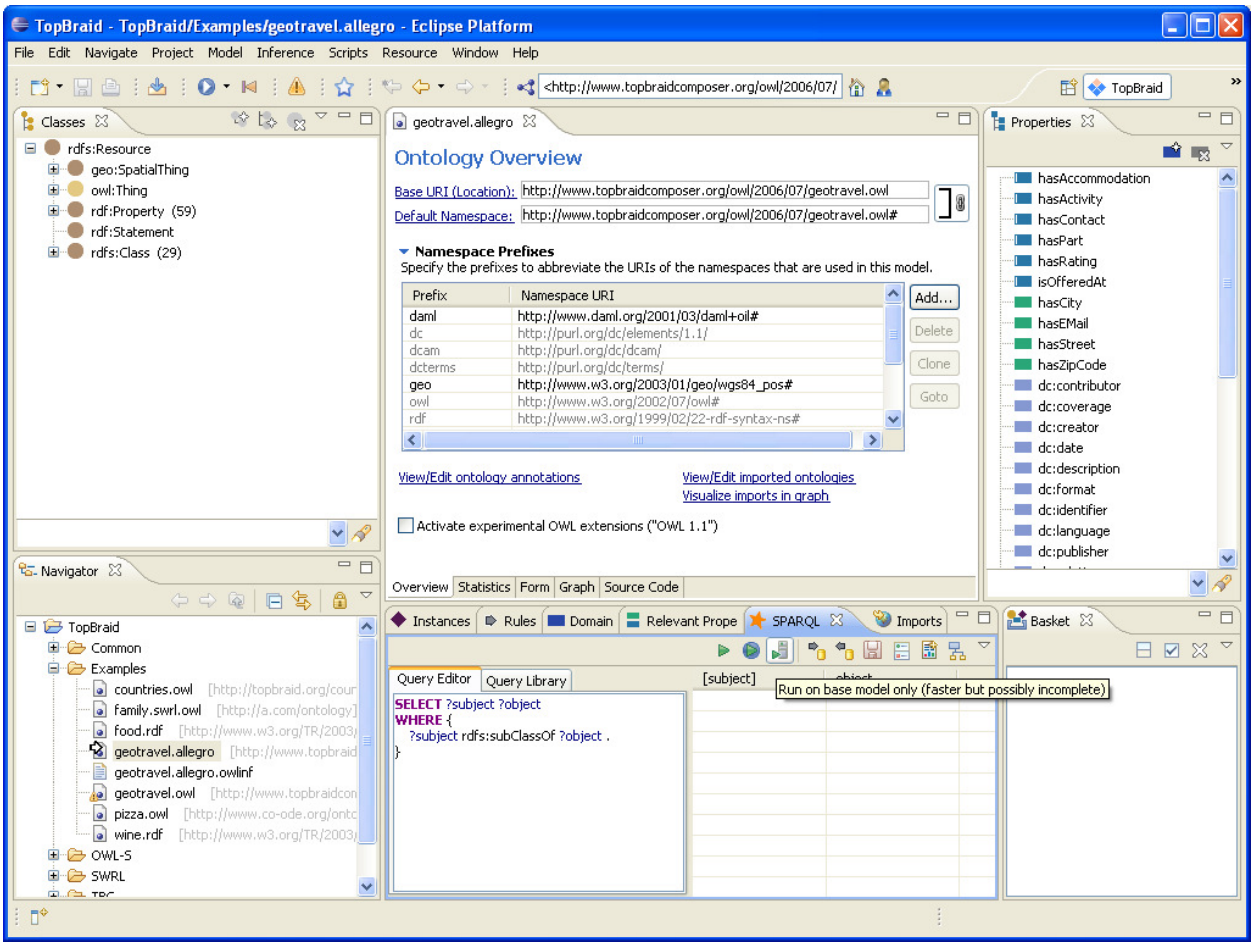
From the "Export into format" select AllegroGraph and name the target file geotravel.



Specify the AllegroGraph database name and the directory that will house the database. The host and port parameters are those that you used to start the AllegroGraphServer. Specify the named graph into which you want the triples exported -- leave it blank for the default graph.

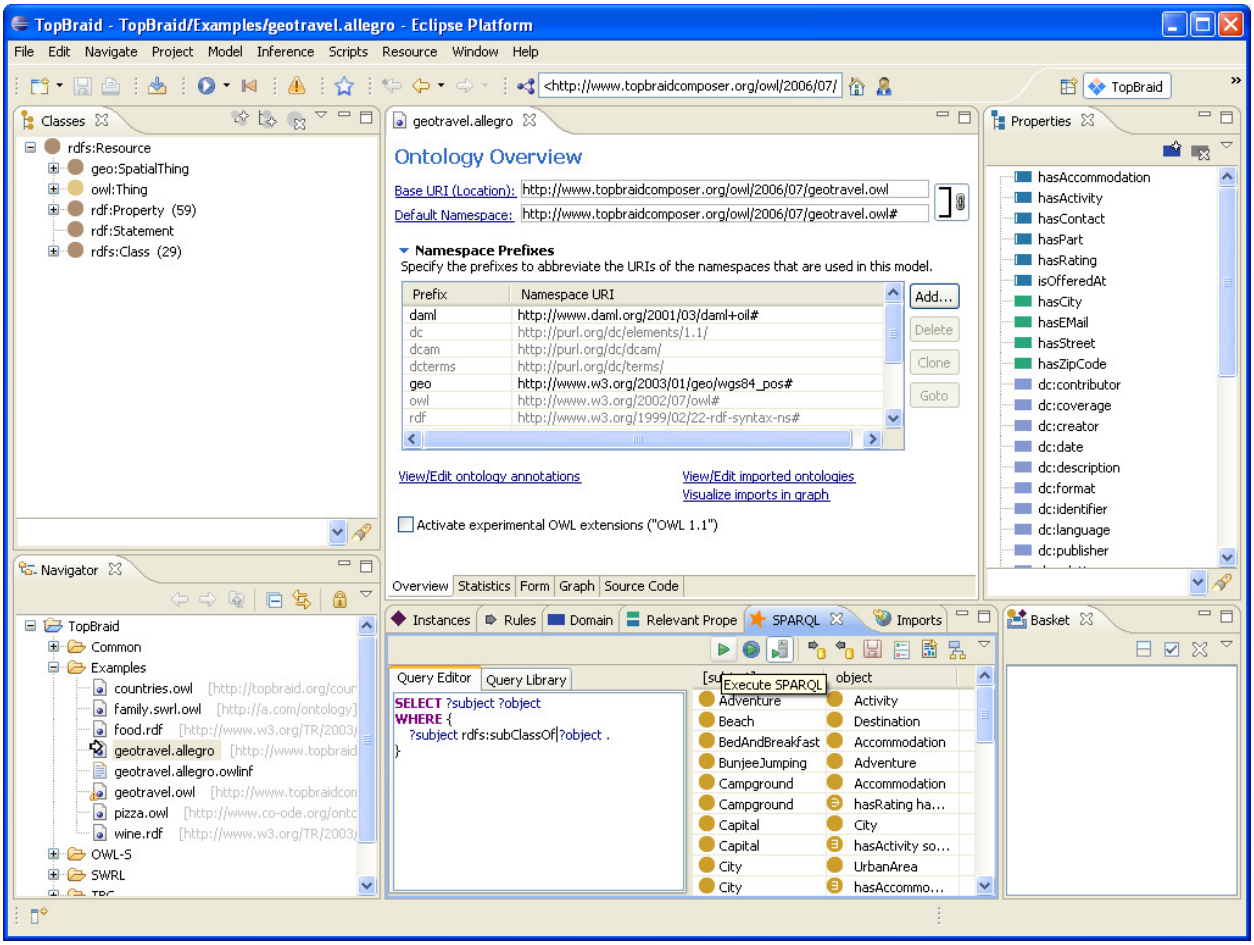


You should now have a file geotravel.allegro in the workspace and a fully indexed AllegroGraph database in /tmp/tbc/geotravel/. Close geotravel.owl and open geotravel.allegro to work with the AllegroGraph database:

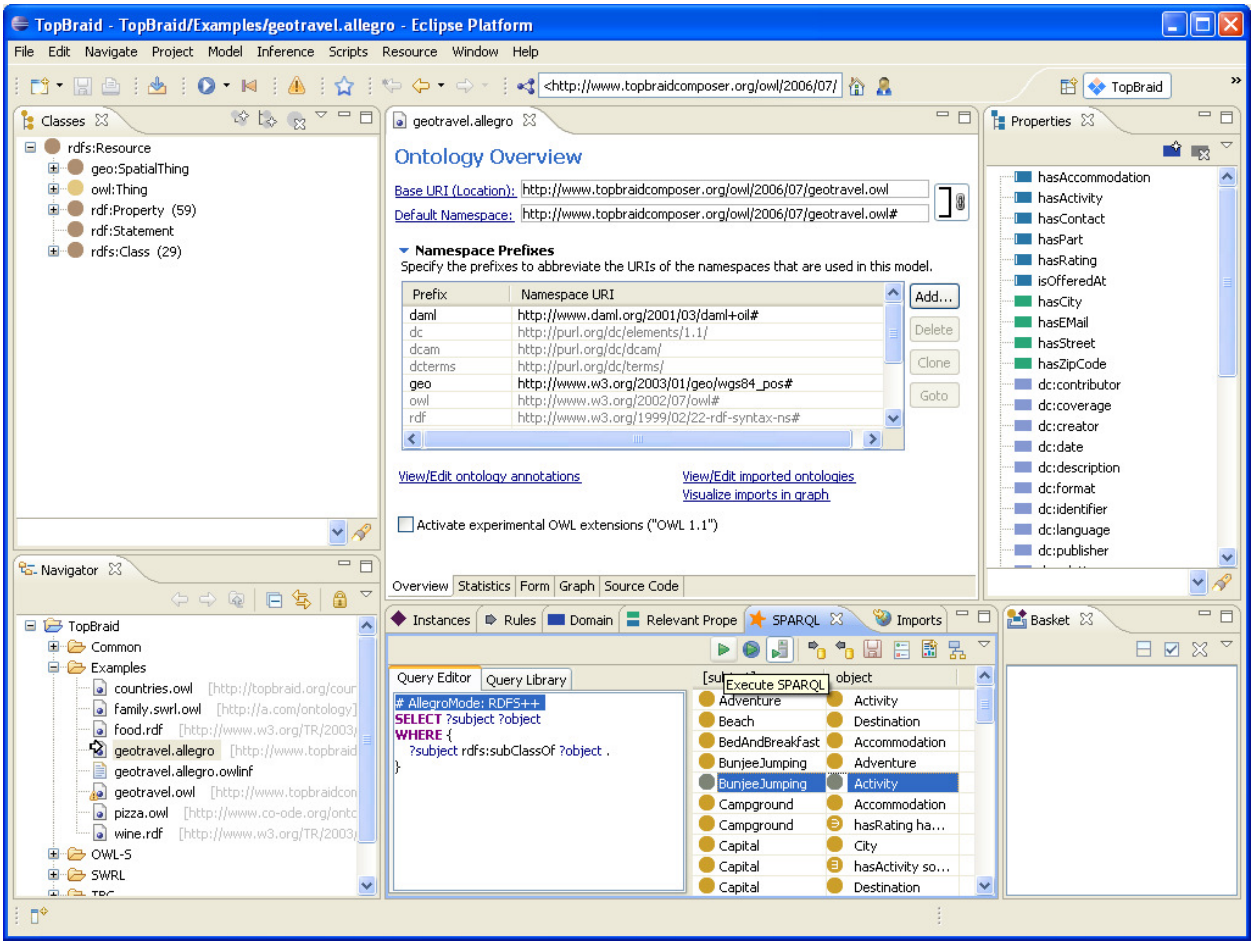


Querying an AllegroGraph database

Select the SPARQL View and ensure that “Run on base model only” is selected (as shown in the image above); this ensures that the query will be executed by the AllegroGraph server rather than the TBC client. Select the Execute button to see the results:



Adding the comment “# AllegroMode: RDFS++” as shown below will include results from AllegroGraph's native RDFS++ reasoner. Note that BunjeeJumping is also now inferred to be a subClassOf Activity:



It is also possible to do ASK and CONSTRUCT SPARQL queries with and without RDFS++ reasoning. AllegroGraph's native Prolog Select queries are also supported by TBC 3.2.