**Common Action 2021 Quantum Hackathon Challenge**

**Using the ESG KGs**

**Graph Database and Log-In Details**

All the data for the Common Action Quantum Challenge is stored in a Stardog graph database[[1]](#footnote-1) which is accessed at https://<some\_url>. Each hackathon team has an account with the following details:

* User name: hankenteam# where the # is 1-15
* Password: ……

The accounts are set up with read/query privileges to the “hanken-esg” database.

**Getting Started**

Enter the graph database URL into your browser and you will be prompted for your user name and password. Once successfully entered, you should see Figure 1.

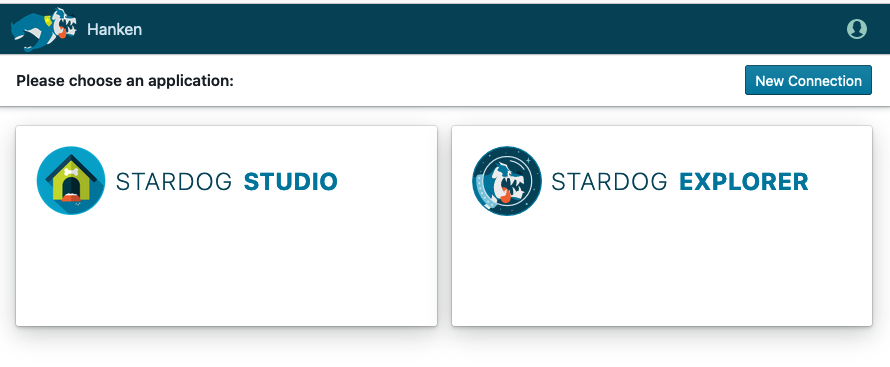


Figure 1. Opening Screen

Click on “Stardog Studio” and Figure 2 will be shown in a new tab. Feel free to take the “tour” to familiarize yourself with the application, or just continue reading. Note that the application places you in the “Workspace” (where queries can be run) by default. The “Workspace” is denoted by the page icon, highlighted in blue, on the left side of the window.

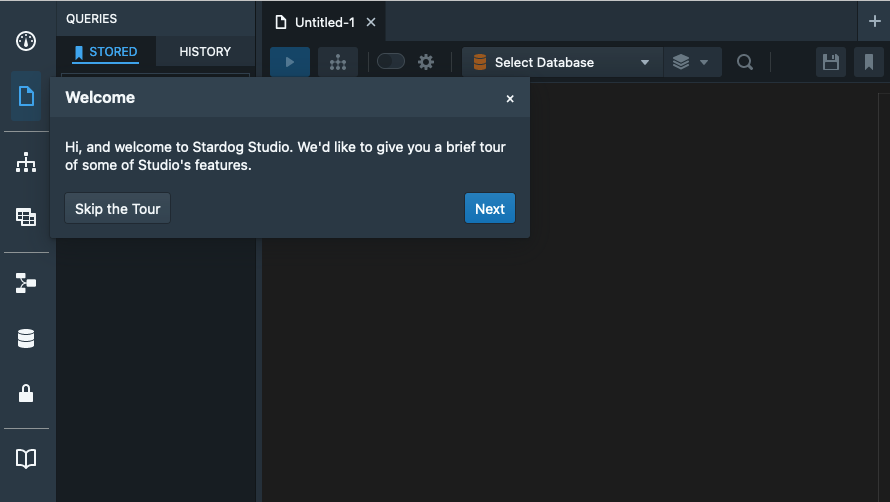


Figure 2. Initial Window of Stardog Studio

Selecting a database is required before anything more can be done. You do this by pressing the down arrow next to the “Select Database” text and choosing “hanken-esg” (your only choice). This is shown in Figure 3.

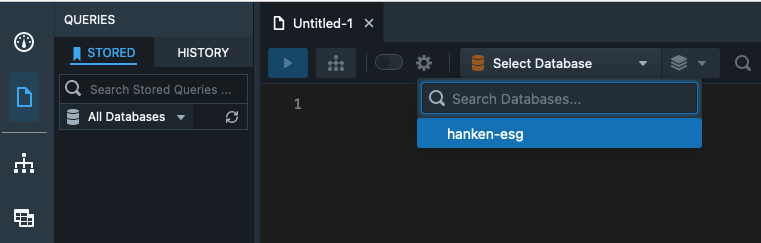


Figure 3. Getting into the Stardog Hackathon Database

Note that after logging out of the database, closing the browser window or being inactive for some time, you will see a slightly modified Studio window. In this case, instead of the “Select Database” prompt, you will see the text, “Connect to Stardog”. When this happens, click on the “Connect” and a pop-up will appear. This allows you to automatically and correctly connect to the hanken-esg database in the future. Set up your connection as shown in Figure 4 (making sure to substitute your team’s user name).

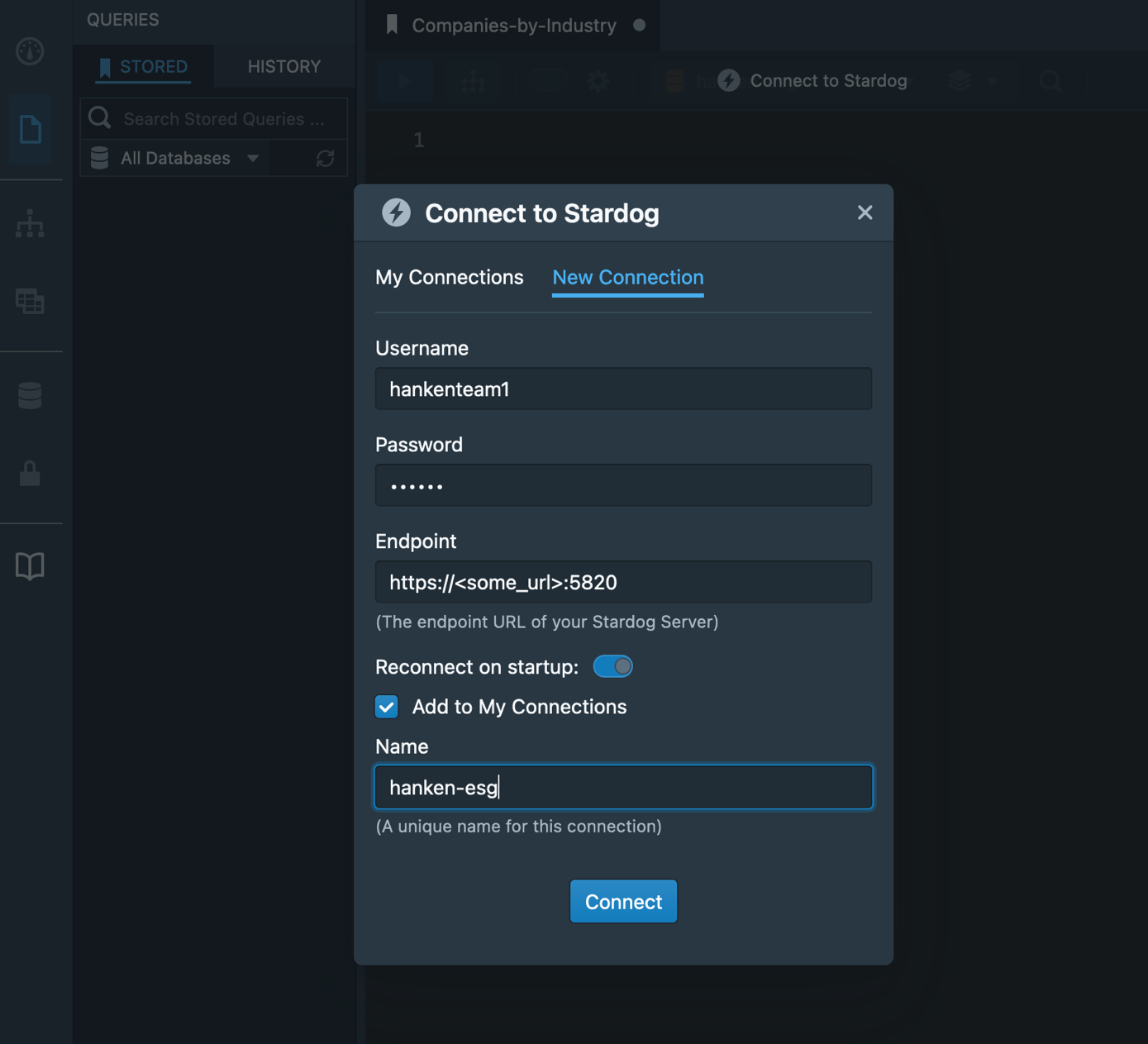


Figure 4. Setting Up an Automatic Connection to the Database

**Running Queries**

Now we are ready to play with the data by executing some queries. Queries are written in SPARQL[[2]](#footnote-2), which is a language very similar to SQL, and yet sometimes very different. As opposed to defining and working across data in rows and columns in rigid tables, SPARQL specifies a series of relationships that should be traversed to find variables of interest.

A set of queries have been predefined and are available for you to customize and execute. However, before digging into the details of these queries, let’s explore one or two in depth to give you some necessary background.

The query shown in Figure 5 is the pre-stored “Companies-by-Industry” query.

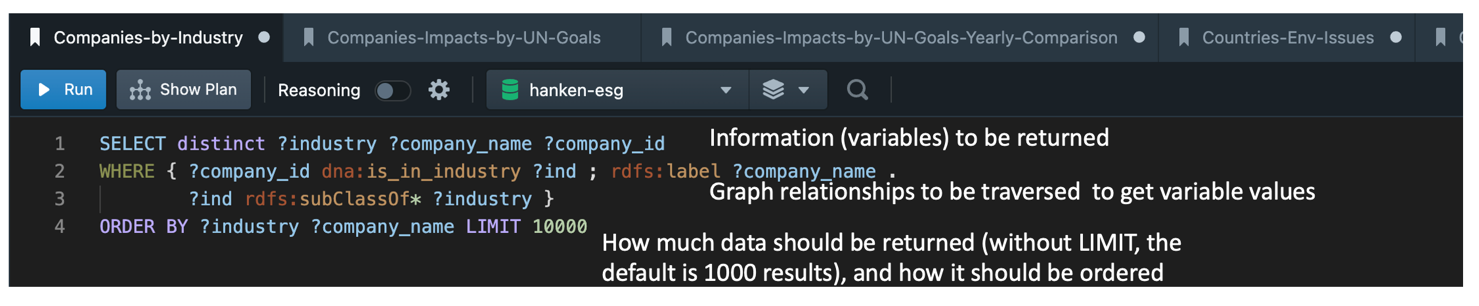


Figure 5. The Companies-by-Industry Stored Query

The second and third lines instruct the query engine to get all entities that have an “is\_in\_industry” relationship to *something*, walk “up” the inheritance hierarchy for that *something*, and retrieve all the entities, and *somethings* and their superclasses.

What does it mean to walk “up” the inheritance hierarchy for industries? Figure 6 shows the first level of industries in the Hackathon data. (This image is taken from the hackathon-ontology-tree.html[[3]](#footnote-3).) Expanding out the tree for “Line of Business” and then for “Military Related Business”, one finds the subclass, “Aerospace and Defense”. Therefore, a company in the Aerospace and Defense industry would also be listed (at least) as a “Military Related Business” and a “Line of Business” (along with a few other industry types, due to multiple inheritance). This allows for working at the level of very specific industries or very general ones.

Note that it would be possible to remove the top-level “Line of Business” class from the results, but that was not done since it can be easily ignored.

It is valuable to point out that all the stored queries can be edited. For example, to only return the most specific industry for a company, change the WHERE clause (the text within the curly brackets) to be “{ ?company dna:is\_in\_industry ?industry }”.

One last thing to discuss is the star (“\*”) sign following “subClassOf”, in the third line. That is a SPARQL property path[[4]](#footnote-4), “a possible route through a graph between two … nodes”. The star indicates that zero or more occurrences of “rdfs:subClassOf” should be retrieved. So, the most specific industry is found using the statement in line 2. Then, that industry and all its superclasses are returned using the statement in line 3.

Results after executing the query in Figure 5 are shown in Figure 7, while the results for the query with the modified WHERE clause (“{ ?company dna:is\_in\_industry ?industry }”) are shown in Figure 8. Huge differences can be seen in the number of results (6446 vs 1901), which is due to walking “up” the inheritance hierarchy. This can be seen in the results since “BankingAndFinanceBusiness” is listed in Figure 7, and then a specific subclass, “BankingBusiness”. But, “BankingAndFinanceBusiness” is not included in Figure 8.

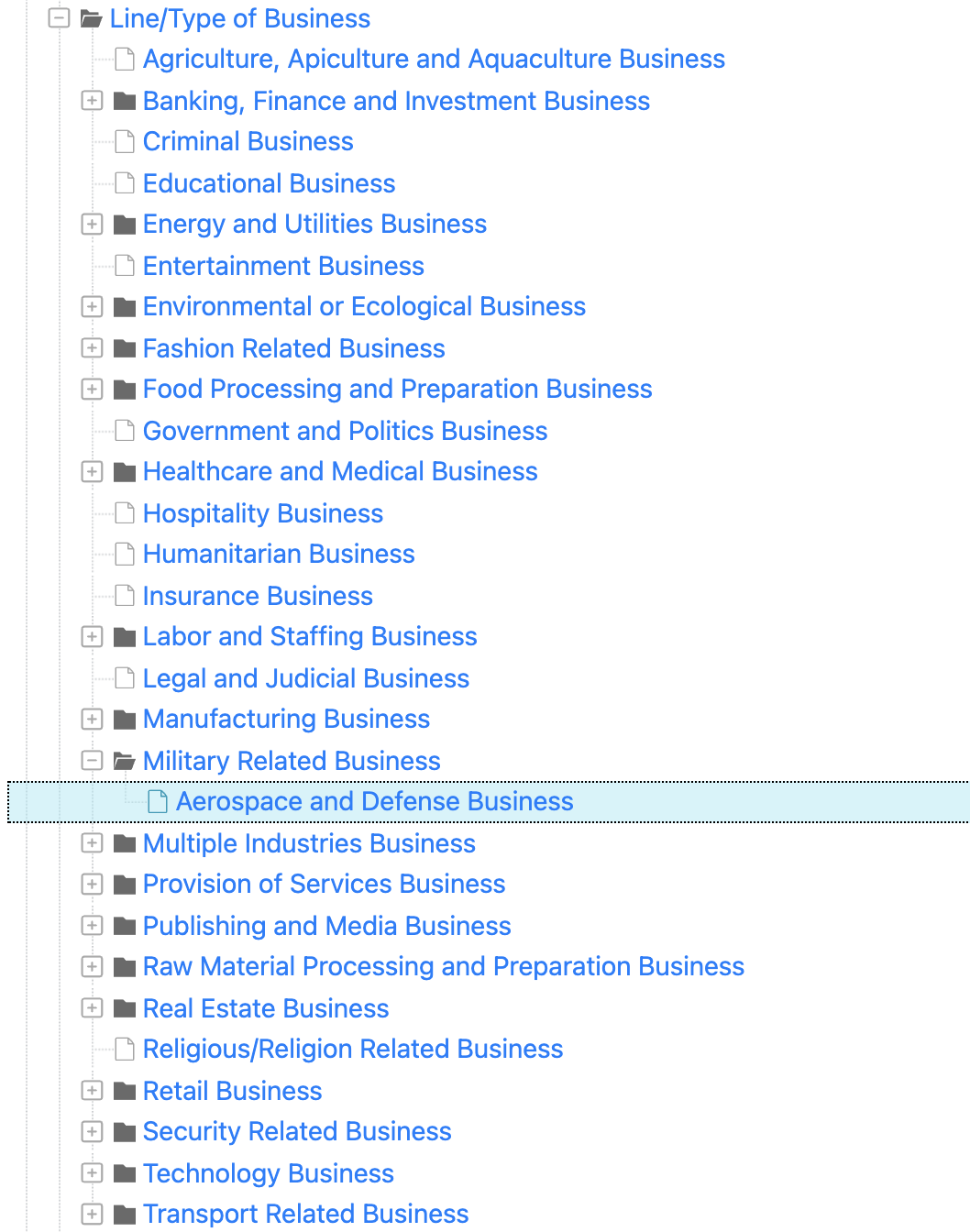


Figure 6. List of Industries/Lines of Business

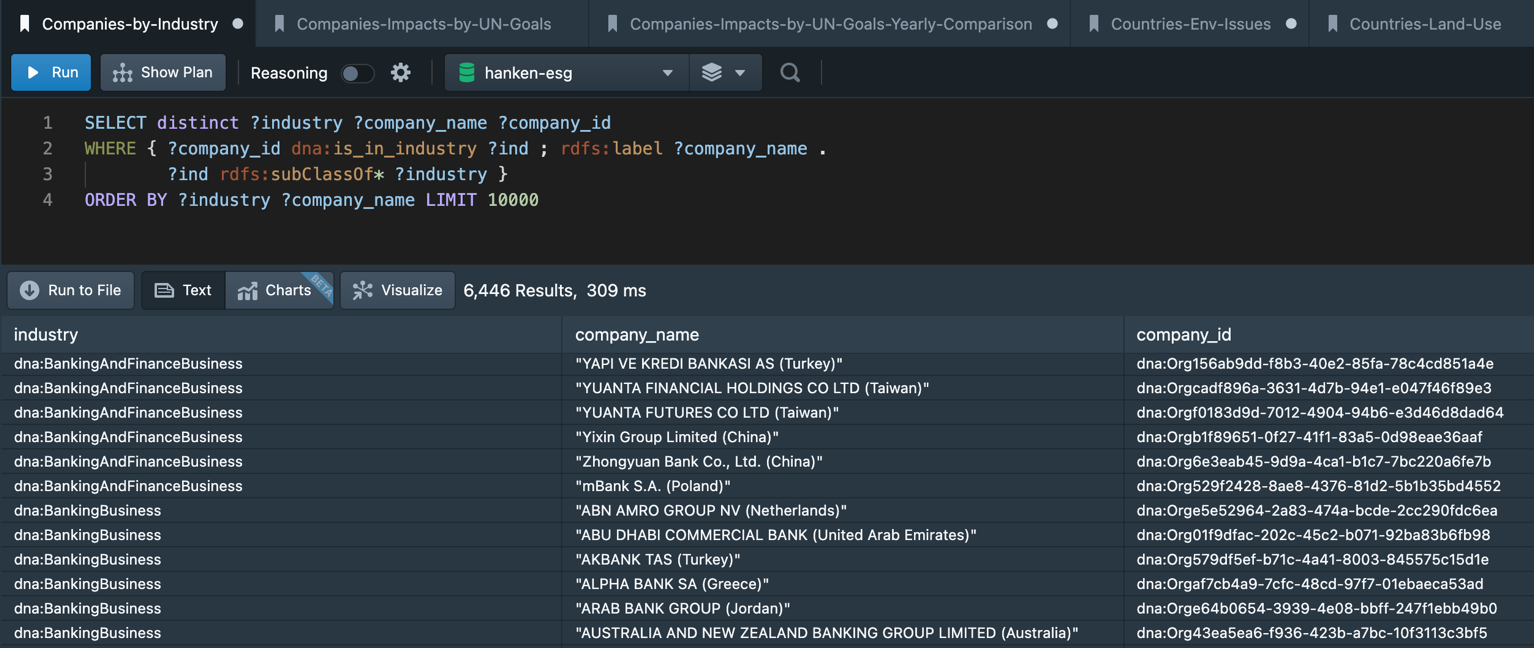


Figure 7. Results from Executing the Companies-by-Industry Query

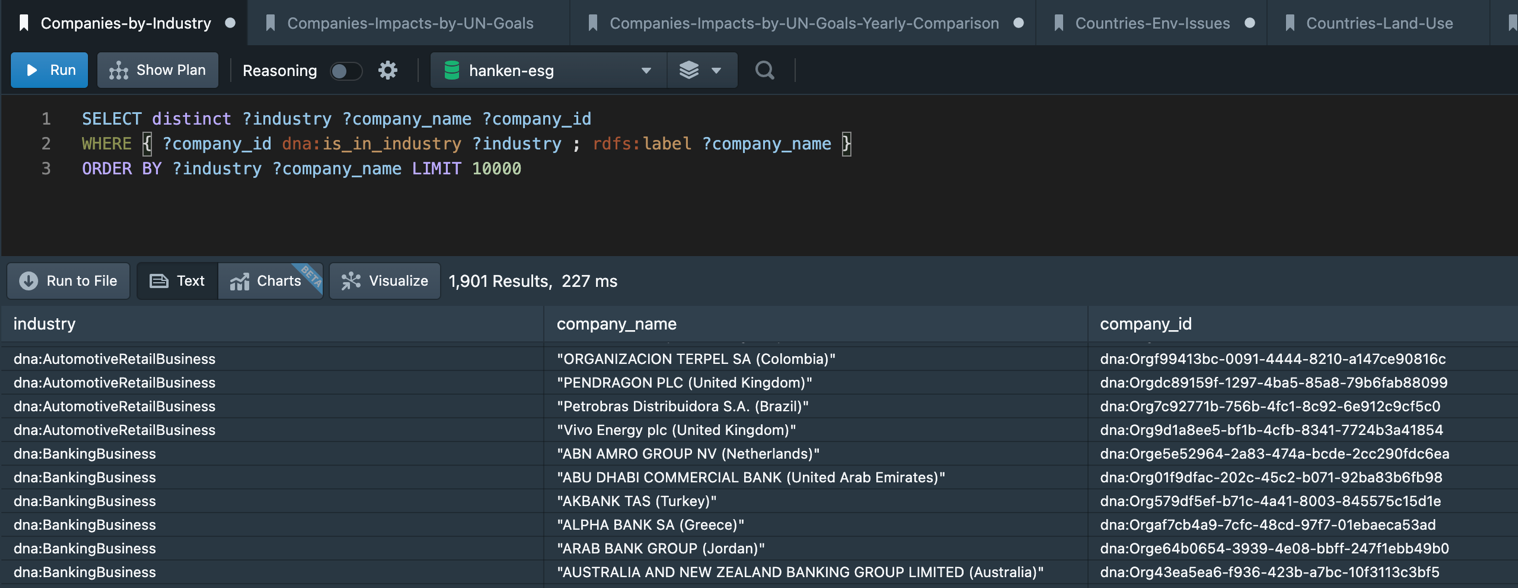


Figure 8. Results from Executing a Modified Companies-by-Industry Query

Another query example explores the use of land in a Country. Figure 9 is the pre-stored “Use-of-Land-Resources-By-Country” query.

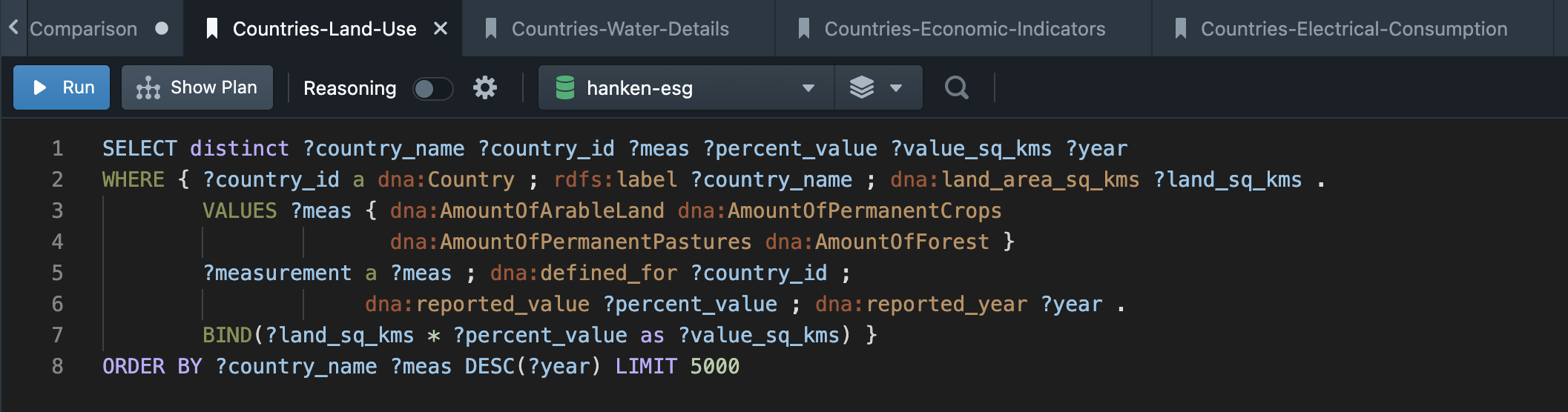


Figure 9. The Countries-Land-Use Stored Query

The query above is formatted similarly to the one shown in Figure 5, but is a bit more complex. Let’s review the details:

* Line 2 indicates that Countries with a land\_area\_sq\_kms reference should be retrieved, as well as the Country name/label
* Lines 3 and 4 indicate the Measurements of interest (all the ones related to land resources)
  + The list of Measurements can be modified to any one or more of the Measurements defined in the spreadsheet, Measurements.xlsx[[5]](#footnote-5)
* Line 5 retrieves all instances of the specified types of Measurements, that are defined for each Country meeting the criteria of line 2
* Line 6 retrieves the “reported\_value” and “reported\_year” for those Measurements
* Line 7 performs a calculation (since the reported\_value is a percentage) that calculates the actual land area
  + The calculated area is retrieved using the variable, “value\_sq\_kms”
  + A calculation is not mandatory since the variable, “land\_sq\_kms” could have been retrieved, with the percentage, and the calculation done separately
  + A wide variety of calculations can be performed such as adding a variable (?area\_sq\_kms) to retrieve the total area of a Country (using the property, “area\_sq\_kms”) and a binding that indicates the area of the Country covered by water (“BIND(?area\_sq\_kms - ?land\_sq\_kms as ?water\_sq\_kms)

Results from executing the query are shown in Figure 10.

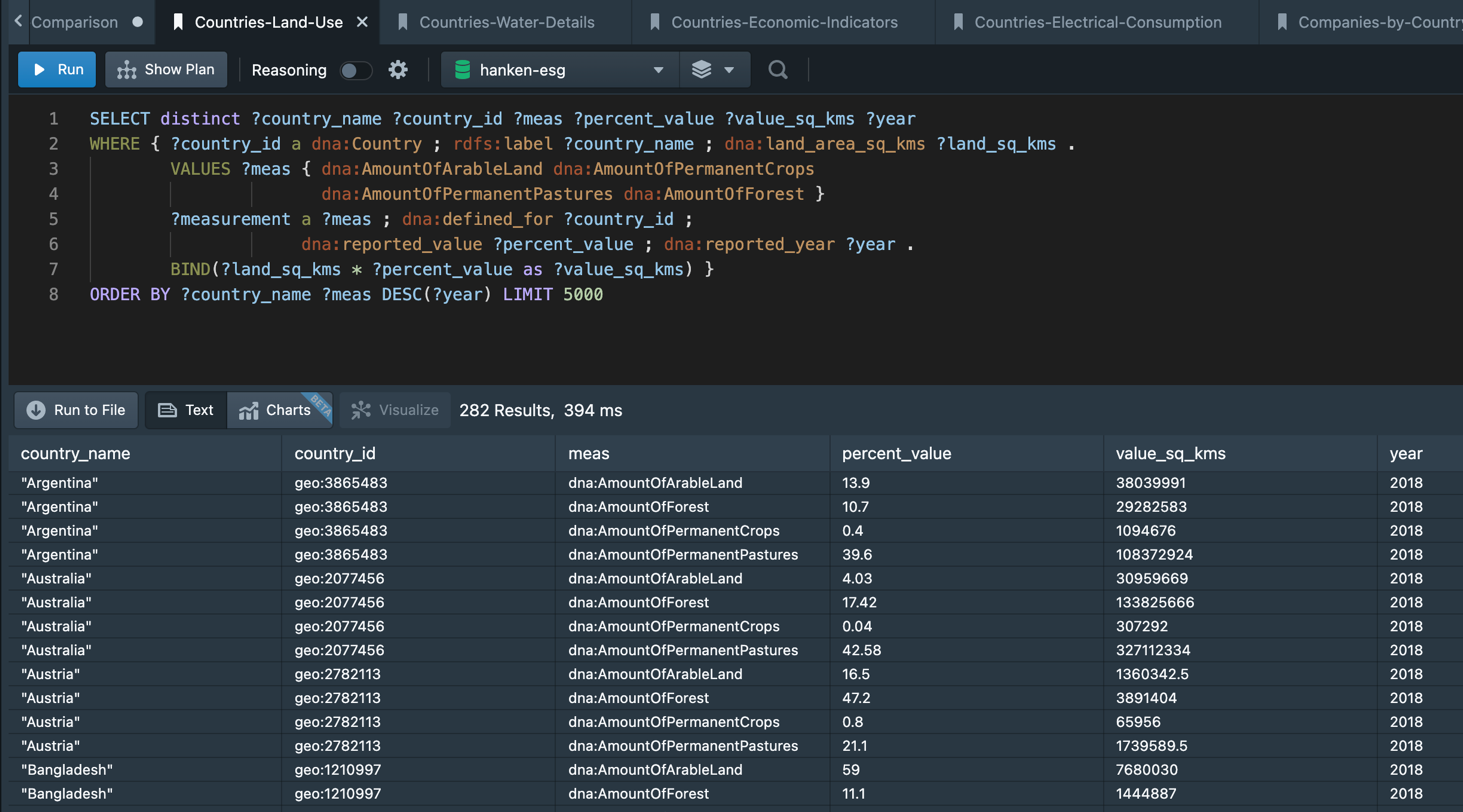


Figure 10. Results from Executing the Countries-Land-Use Query

Queries are executed by first selecting one from the “Stored” tab on the left side of the window. The list of queries (as of Wednesday, Dec 1) is shown in Figure 11. Then, click the blue, “Run” button in the upper-part of the window. Results will be displayed in the lower right. If desired, the results can be downloaded to a file (the comma-separated or “csv” format is recommended) by clicking on the “Run to File” button positioned just above the results. The downloaded file can be loaded as a spreadsheet and further analyzed.

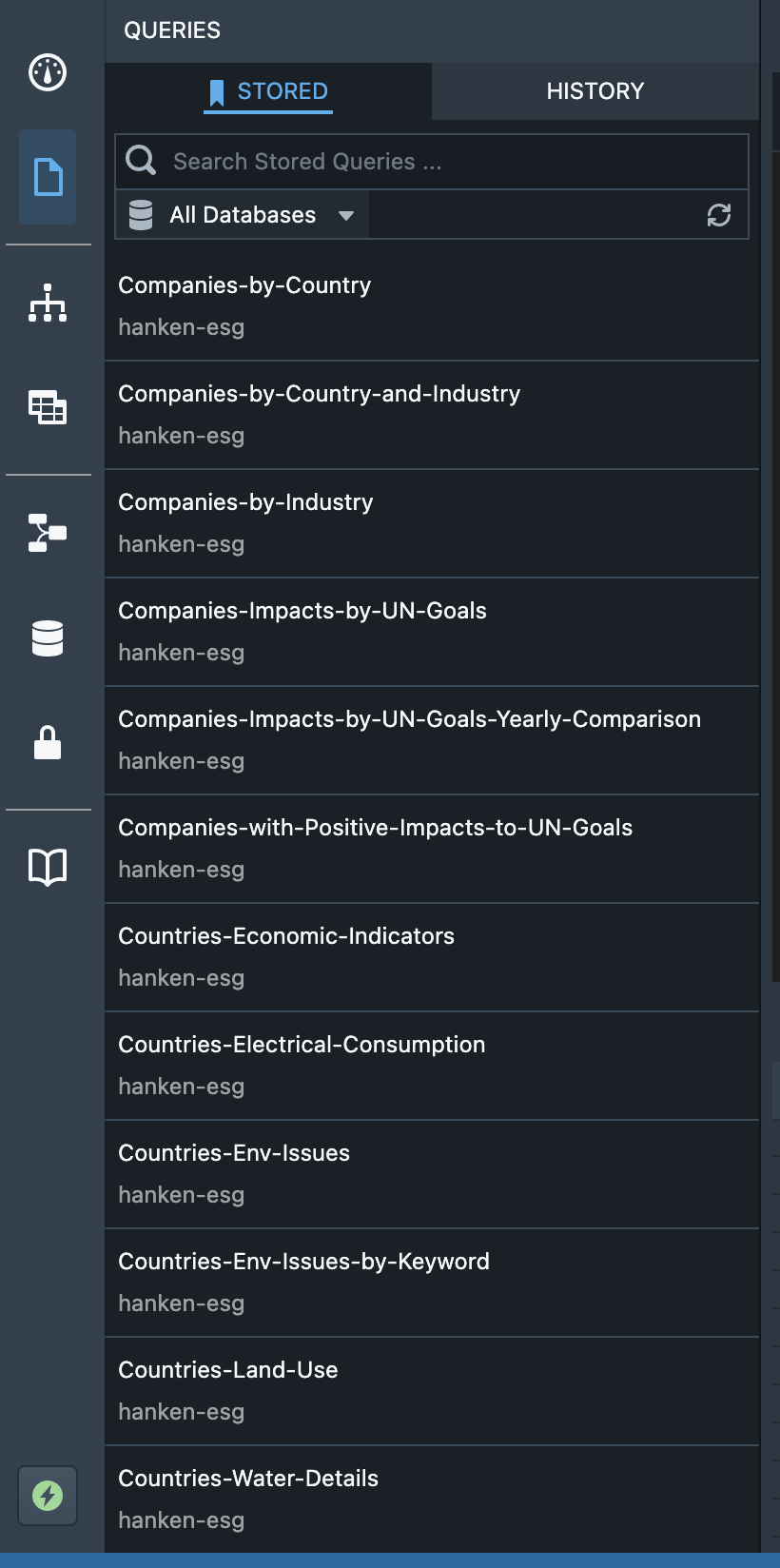


Figure 11. List of Stored Queries

**Inventory of Stored Queries**

The stored queries defined as of December 1 2021 are the following:

* Companies-by-Country – Reports Organizations by their headquarters’ Country
* Companies-by-Country-and-Industry – Reports Organizations by their headquarters’ Country and industry/line of business
* Companies-by-Industry – Reports Organizations by their industry/line of business
* Companies-Impacts-by-UN-Goals – Reports each Organization’s impacts (positive or negative) on various UN Goals, by year
* Companies-Impacts-by-UN-Goals-Yearly-Comparison – Compares an Organization’s impact on a UN Goal, year-to-year, and orders the results to show the Organizations with the most positive improvements across 2 years, and the years when that occurred
* Companies-with-Positive-Impacts-to-UN-Goals - Reports only positive impacts to UN Goals for an Organization, and orders the results by impact value
  + The year of that impact is also
* Countries-Economic-Indicators – Reports Countries’ GDP, labor force, unemployment, … details
* Countries-Electrical-Consumption – Reports Countries’ electricity production/consumption and generating capacity, as well as % electrical capacity from fossil fuels, nuclear fuels and renewable sources
* Countries-Env-Issues – Reports Countries environmental issues as defined by the CIA World Factbook
* Countries-Env-Issues-by-Keyword – Allows search of the Countries’ environmental issue text by text strings
  + Possible text for search is defined at <https://www.cia.gov/the-world-factbook/field/environment-current-issues>
* Countries-Land-Use – Reports Countries’ land area size and the percentages of land dedicated to crops, pastures and forests
* Countries-Water-Details – Reports Countries’ water area size (which includes all inland water bodies, rivers and reservoirs) and the total amount of available, renewable water resources

**Stardog Explorer**

This paper has been mainly focused on querying the data using Stardog Studio. However, Figure 1 also shows another tool, Stardog Studio. Studio is basically a search tool to retrieve class and instance information from the database, based on searching the entity labels.

To use Explorer, type in one or more words, and hit Enter. A sample search for “United States” returns the results shown in Figure 12.

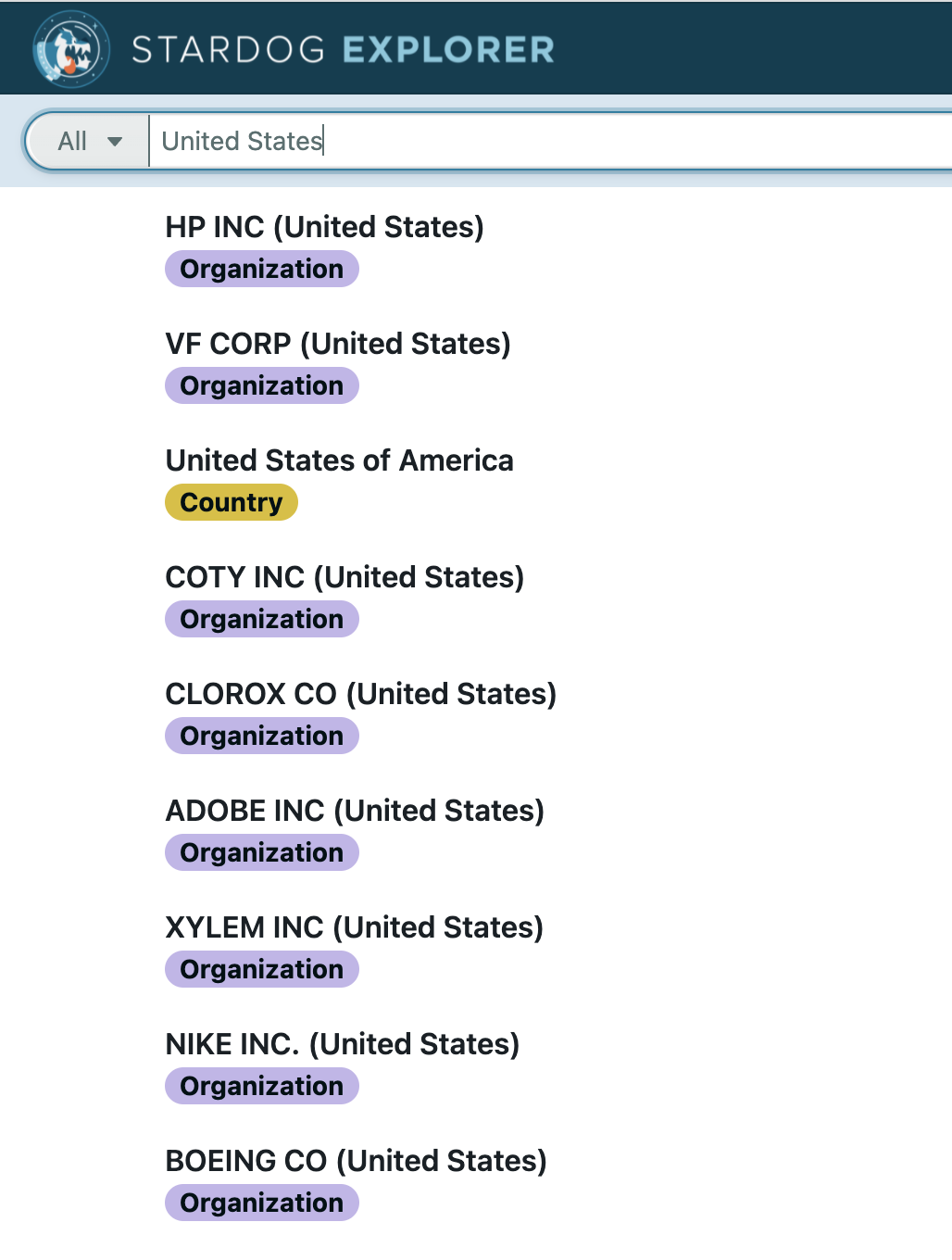


Figure 12. Results of Search Using Stardog Explorer

1. <https://www.stardog.com/> [↑](#footnote-ref-1)
2. <https://en.wikipedia.org/wiki/SPARQL> [↑](#footnote-ref-2)
3. The tree HTML can be downloaded from <https://github.com/ontoinsights/deep_narrative_analysis/blob/master/ontologies/hackathon-extensions/hackathon-docs/hackathon-ontology-tree.html?raw=rue> [↑](#footnote-ref-3)
4. <https://www.w3.org/TR/sparql11-property-paths/> [↑](#footnote-ref-4)
5. The spreadsheet can be downloaded from <https://github.com/ontoinsights/deep_narrative_analysis/blob/master/ontologies/hackathon-extensions/hackathon-docs/Measurements.xlsx?raw=true> [↑](#footnote-ref-5)