Enriching Data

In this exercise we are going to use data from opencorporates to enrich a government financial transactions dataset. The purpose of this exercise is to real the power of linked data and how you can go about creating linked data from tabular, transactional data.

The dataset we are going to be using is the Foreign and Commonwealth Office spend over £25,000 dataset.

http://data.gov.uk/dataset/financial-transactions-data-fco

In the making of the exercise the **April 2010** dataset was used.

The question we are going to try and answer is:

"How many of the companies the FCO dealt with in ….. are still active, how many are dissolved and how many have been liquidated?"

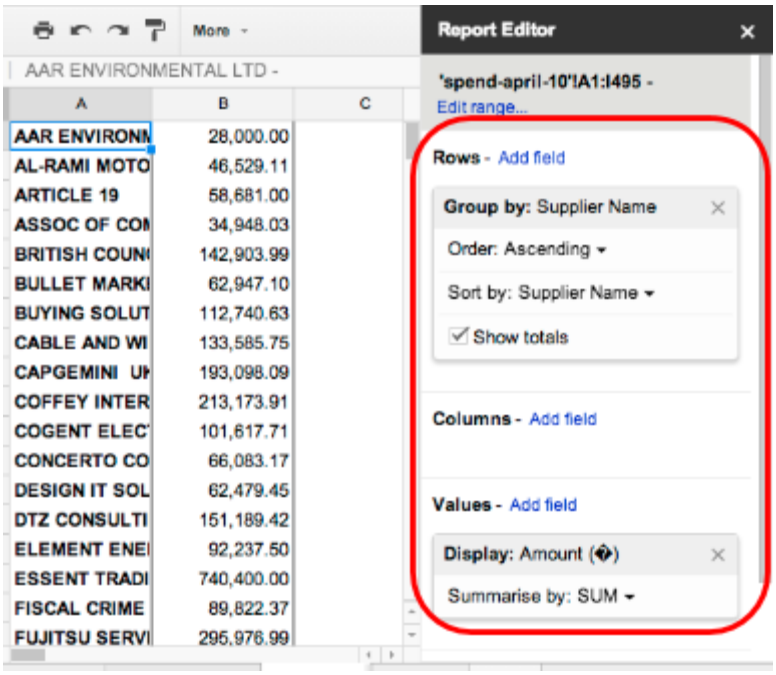
Step 1 - Download and Pivot

For this exercise we do not need all the detail of the transactions. What is required is the total amount spent with each company. To filter and group the data we can use a pivot table, available via google docs (spreadsheet).

Once you have downloaded the data, upload it to google docs and create a google sheet. Once loaded, from the **data** menu select **pivot table report**.

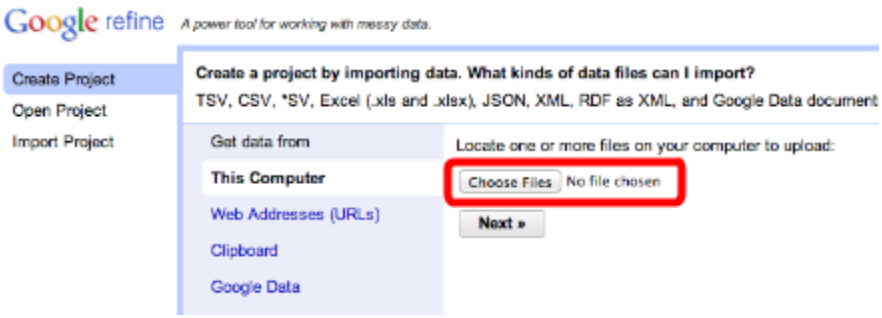
To create the pivot table which will group the data by company name, select the options specified on the right.

Once done export this pivot table as csv.

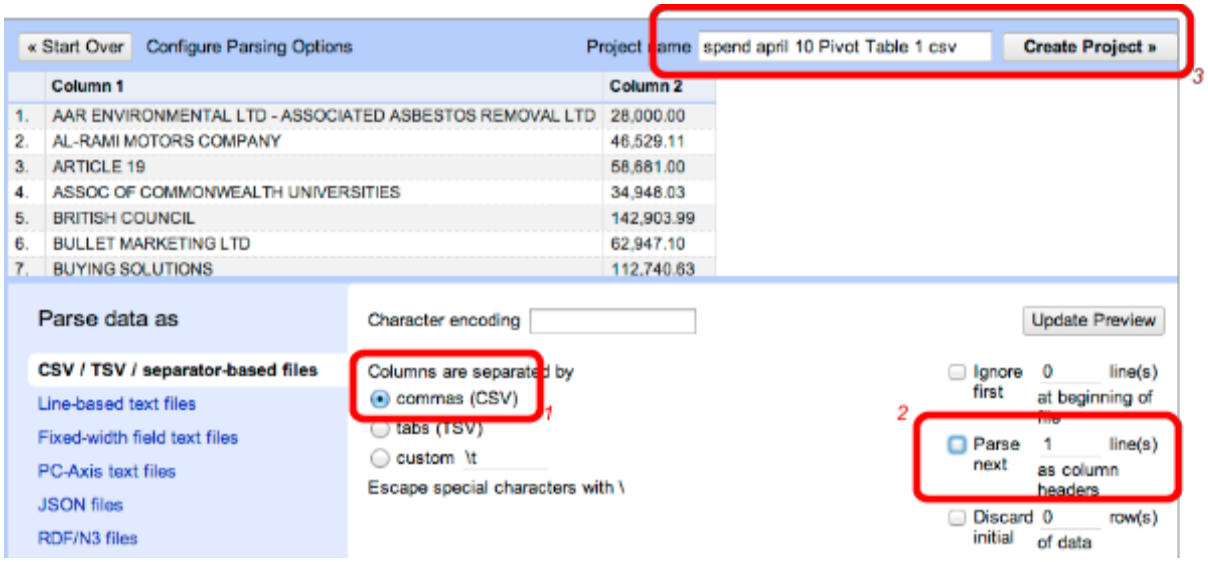


Step 2 - Import into Refine

Having obtained an aggregated version of the data with many fewer rows, import this data into Google/Open Refine.

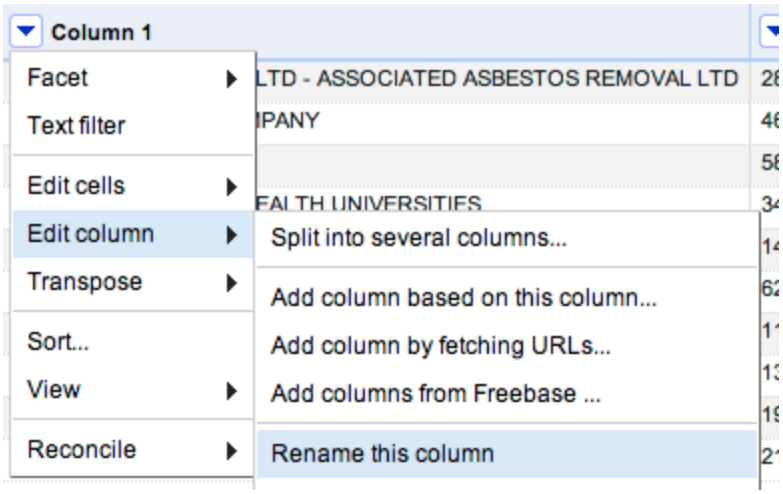


When you import the data, ensure that you do not use the first row as the column headers. The exported pivot table from google docs won't have any.



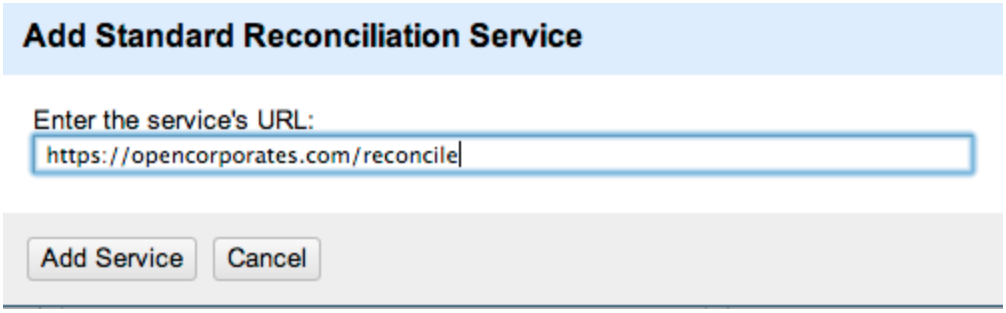
Step 3 - Reconcile and link

With the data now imported we have a column listing company names in plain text and amounts spent with each. Before we continue you might want to change the column titles to reflect the data.



In order to link the data to publicly available data we are going to use the opencorporates reconciliation API. From you company name column drop down select the **Start Reconcile** option from the **Reconcile** menu. You will note that opencorporates is not yet available as an option so a new **standard reconciliation service** will need to be added. The service url is:

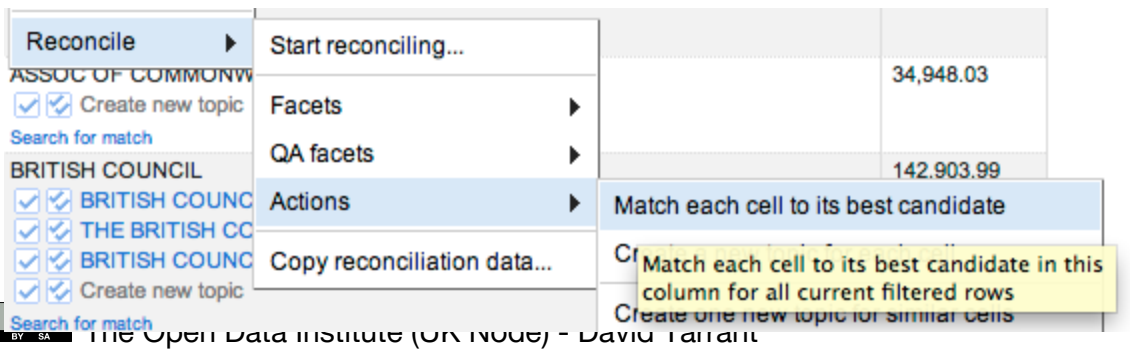
`https://opencorporates.com/reconcile`



Once done, select this service and click **Start Reconciling.**

When complete you will notice that each company may have several matches in opencorporates, with each scored differently. You can click on each option to see a brief overview of that company and then tick which you believe is the correct option.

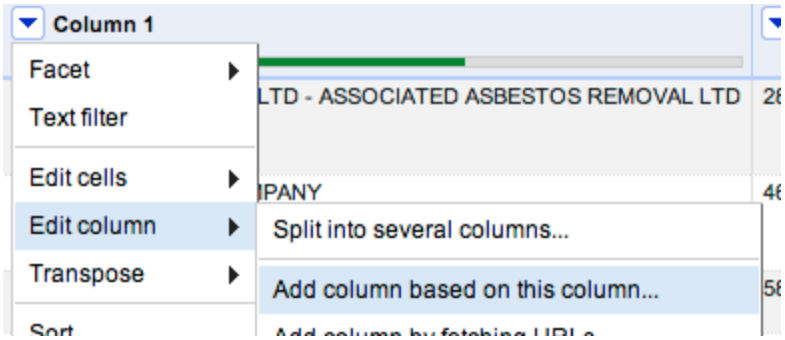
Rather than manually process the entire dataset you can also filter to high or low quality matches using the facet on the left. You can also just choose to match all companies against their best candidate using via the **match** option in the **reconcile** menus **actions**.



Step 4 - Reveal the URI

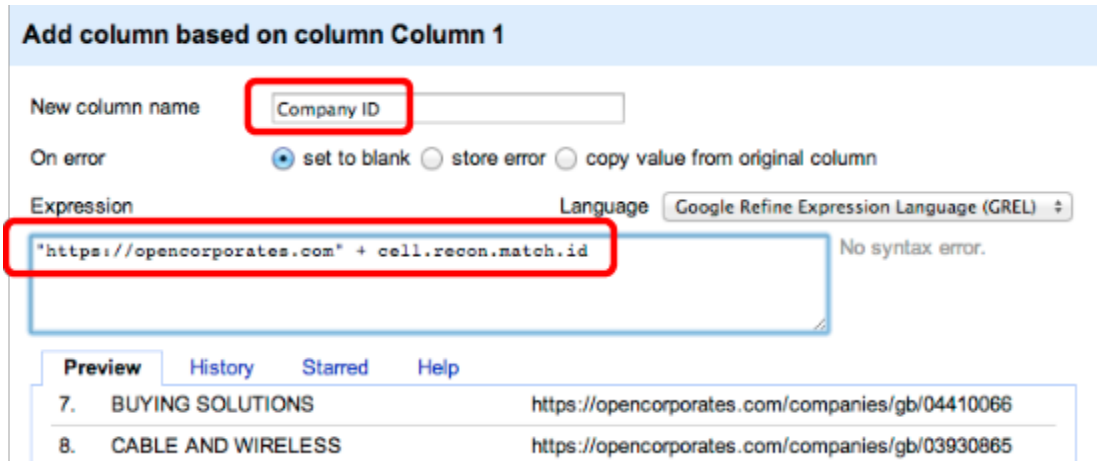
Once you have matched as many companies as possible you will notice that clicking on the company will now take you to that companies page in opencorporates. This URL is also the company URI (identifier) and we can also reveal this in the dataset as follows.

From the company column drop down select **Add column based upon this column**.



In the box that appears, give the column a name of **Company ID** and type the following in the expression box:

```
"https://opencorporates.com" + cell.recon.match.id
```



When done click OK. We could now export this dataset as CSV, XML, RDF etc and it would contain a URI from which further data could be obtained. This is a simple but effective example of linked data and linked csv.

Step 5 - Get the data

We now have a link to the opercorporates page for each company that lists data about that company. It is the current status that we need in order to answer the original question. In order to fetch it for the companies we shall use the opencorporates API in order to download the data for each company.

This stage involves adding a **column by fetching URLs**. Although we could fetch the URI from step 4 using content negotiation, refine does not understand HTTP redirection thus we would not get the data back from the URI. For this reason it is necessary to cut a corner and download the data directly from where the redirect would have sent refine.

```
"https://api.opencorporates.com" + cell.recon.match.id + ".json"
```

Make sure you name the column and turn down the **throttle delay** such that this process completes in reasonable time. The throttle delay is there so we don't overwhelm an API with requests.

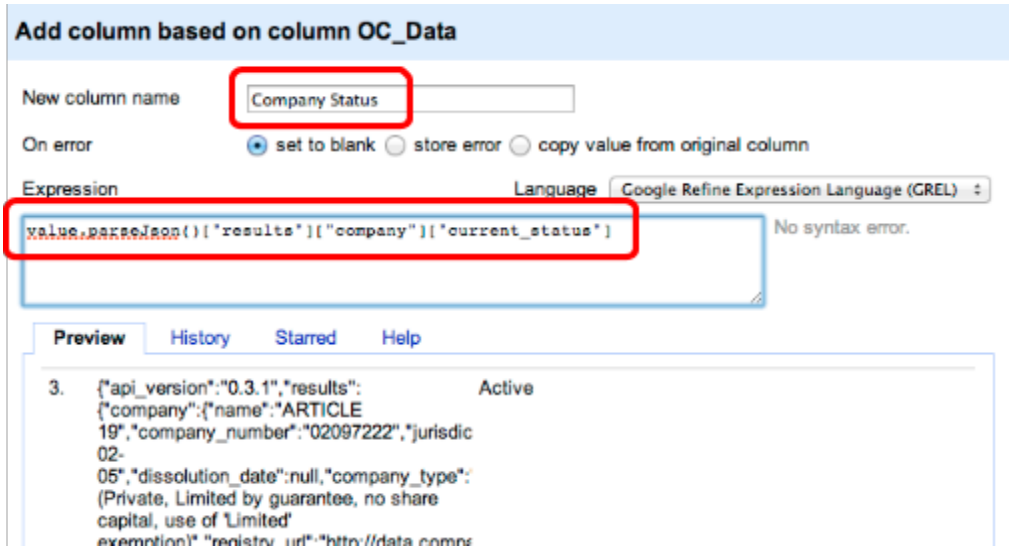
Once done click OK and you should end up with a column full of JSON data from which we need to pick out the **company status**.

In order to see what the JSON data looks like why not take one of the returned values and paste it into the JSON validator at **jsonlint.com**.

Step 6 - Extract the company status

One last column to add based upon the data column this time with the following expression to parse the JSON data:

```
value.parseJson()["results"]["company"]["current_status"]
```



Once complete you might want to **collapse** the data column so that you can see more rows on the screen. It is then possible to apply facets to find out how many companies

are still active, liquidated, dissolved or other.

Extension Exercises

Why not try extracting data other than current status from the opencorporates data?

Can you link to any other data available from opencorporates or suppliers of other reconciliation endpoints?

How about a visualisation of the data?