Atlas Cohort Exchange – Admin

# Introduction

The Atlas Cohort Exchange consists of a set of steps to allow a Cohort definition in a central Atlas instance to be distributed to remote Atlas instances, where the Cohort definition can be re-created, and the inclusion report generated for the local data set. These cohort inclusion report results can then be sent back to the central Atlas instance, where these results can be imported for review.

There are several steps in this process:

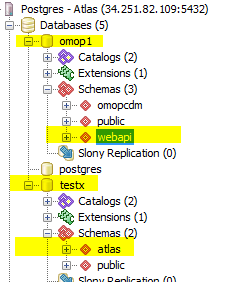
1. Create/define cohort in the central Atlas instance.
2. Export cohort definition as a JSON file from the central Atlas instance using the [AtlasCohortExchange-CohortDefExport](https://github.com/solventrix/AtlasCohortExchange_Notebooks/blob/master/AtlasCohortExchange-CohortDefExport.ipynb) Jupyter notebook.
3. Send the JSON with the cohort definition to remote Atlas site(s).
4. Create the same cohort definition at remote Atlas site using the JSON with the cohort definition.
5. Create inclusion report for the cohort at the remote Atlas site.
6. Export cohort inclusion report results to JSON file using the [AtlasCohortExchange](https://github.com/solventrix/AtlasCohortExchange-ResultImport/blob/master/AtlasCohortExchange-ResultImport.ipynb) R package at the remote Atlas site.
7. Import cohort inclusion report results into central Atlas instance using the [AtlasCohortExchange-ResultImport](https://github.com/solventrix/AtlasCohortExchange_Notebooks/blob/master/AtlasCohortExchange-ResultImport.ipynb) Jupyter notebook.

This document describes the steps to be performed at the central Atlas site (*steps 1, 2, 3, and 7 above*).

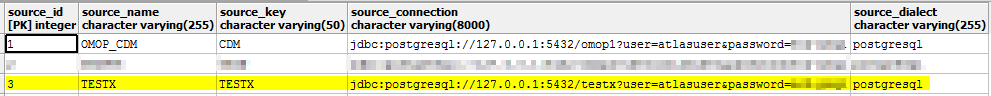
# Prerequisites

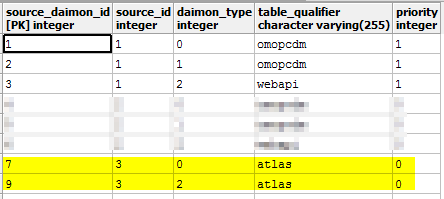
In order to be able to import cohort inclusion report results from external Atlas sites, each of these sites needs a separate schema in the central Atlas database. Basically, each external Atlas site needs a ‘webapi’ database schema accessible via WebAPI at the central Atlas site.

The following is an example setup for a ‘Test X’ external Atlas site, where the central Atlas site’s ‘webapi’ table structure has been duplicated with a different database (‘testx’) and schema name (‘atlas’) on the same PostgreSQL server:



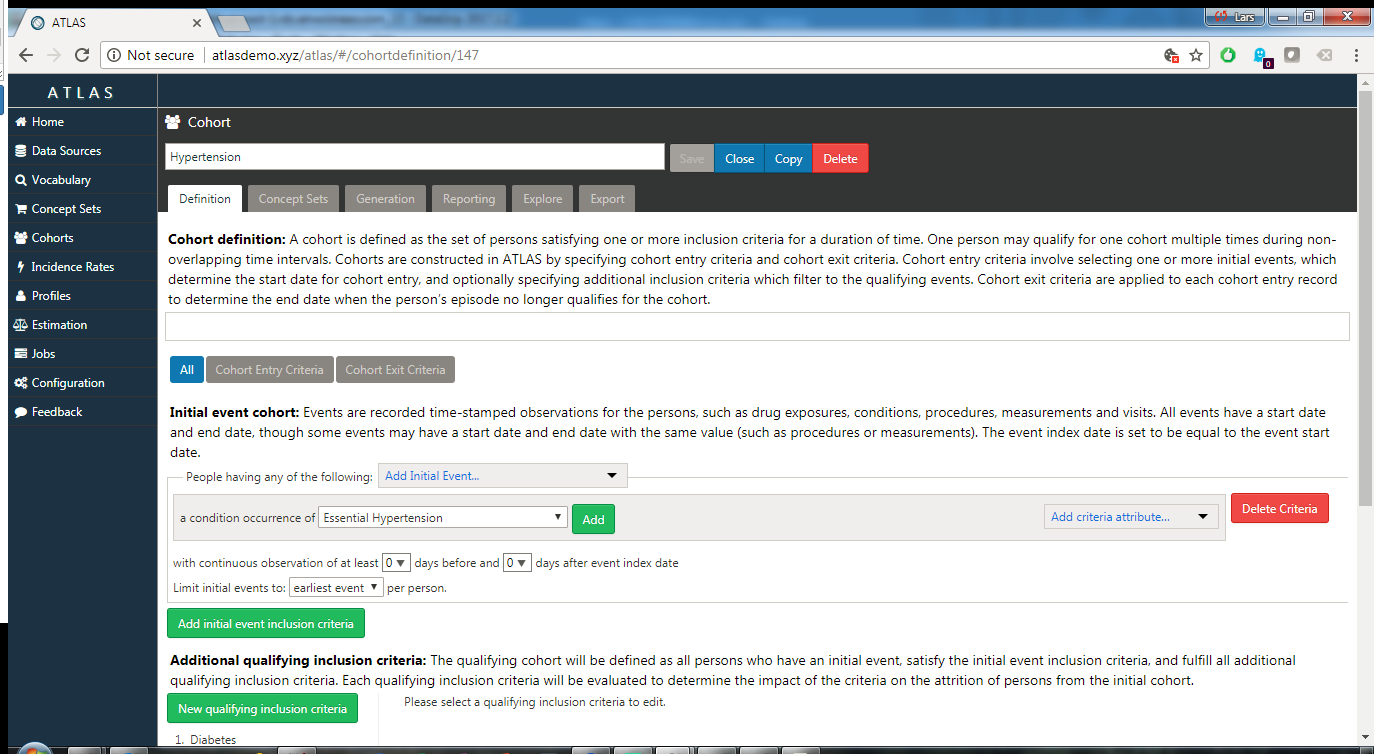
The ***source*** and ***source\_daimon*** entries for this setup is as follows:



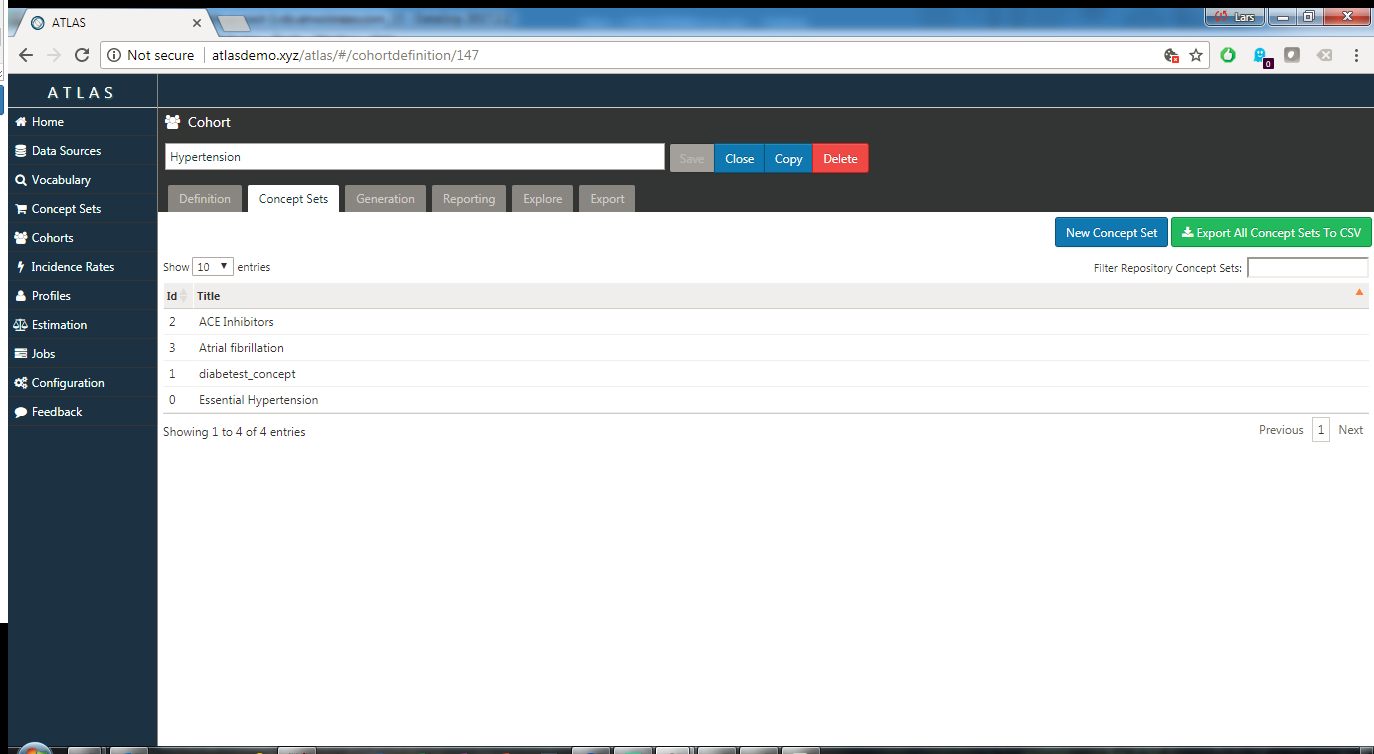


# Export cohort definition JSON and send to external Atlas site

In the central Atlas instance, open the **Cohort** definition for which to send the definition to the external Atlas site(s):

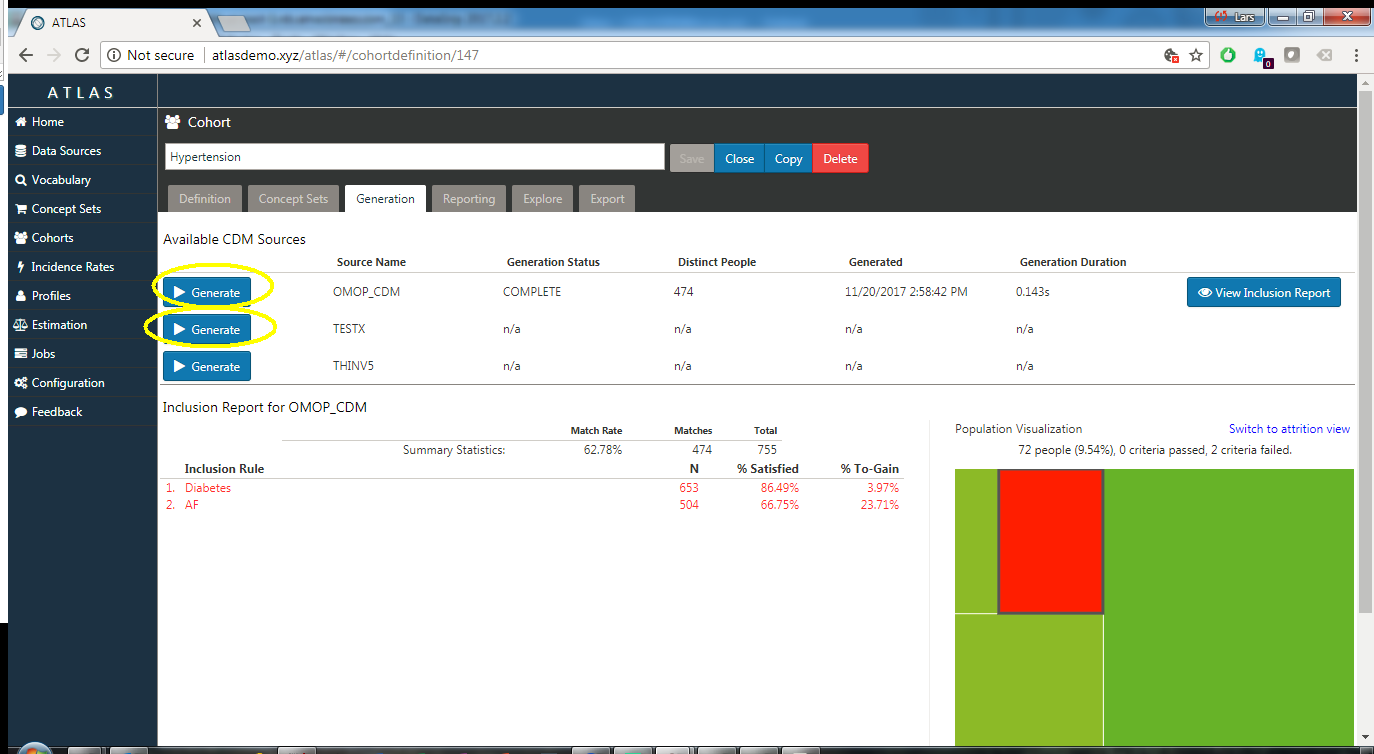


Review the definition if and as needed:



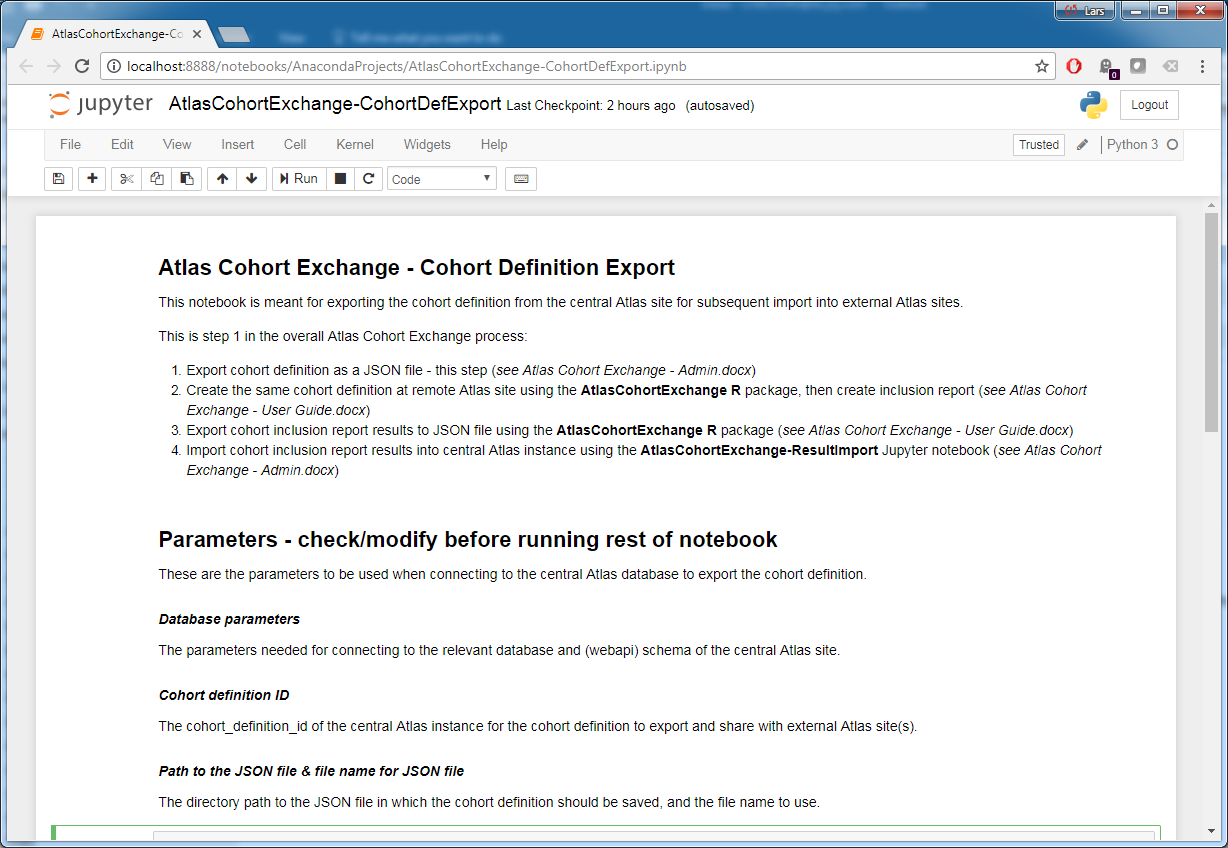
Under the **Generation** tab, make sure to click the **Generate** button for both the local (main) data source as well as for the data source(s) to which the Cohort definition will be sent.

For the local (main) data source, the inclusion report will be generated; for the external Atlas site(s), and empty inclusion report will be generated:

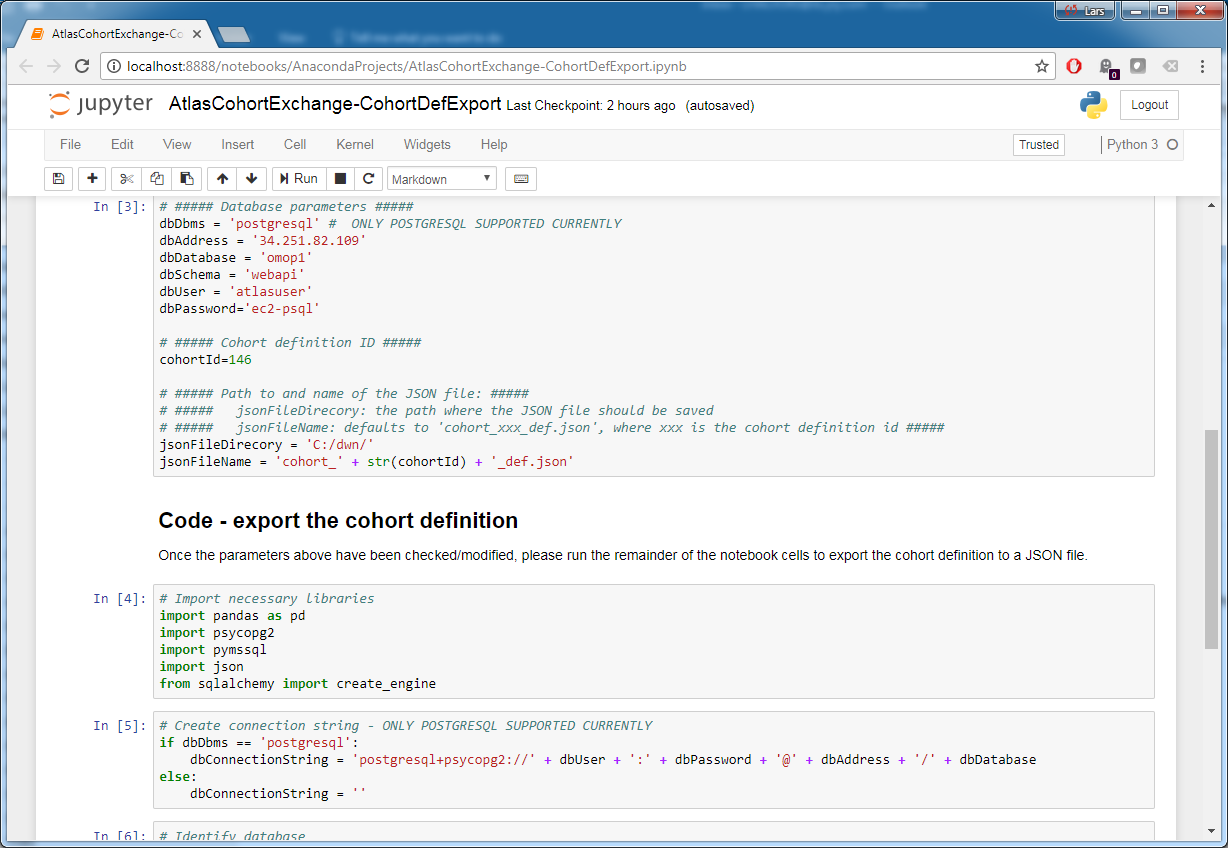


**NOTE**: it is important that the empty inclusion report(s) be generated for the external Atlas site(s) at this point, as they server as placeholders for the inclusion report results that will be imported later.

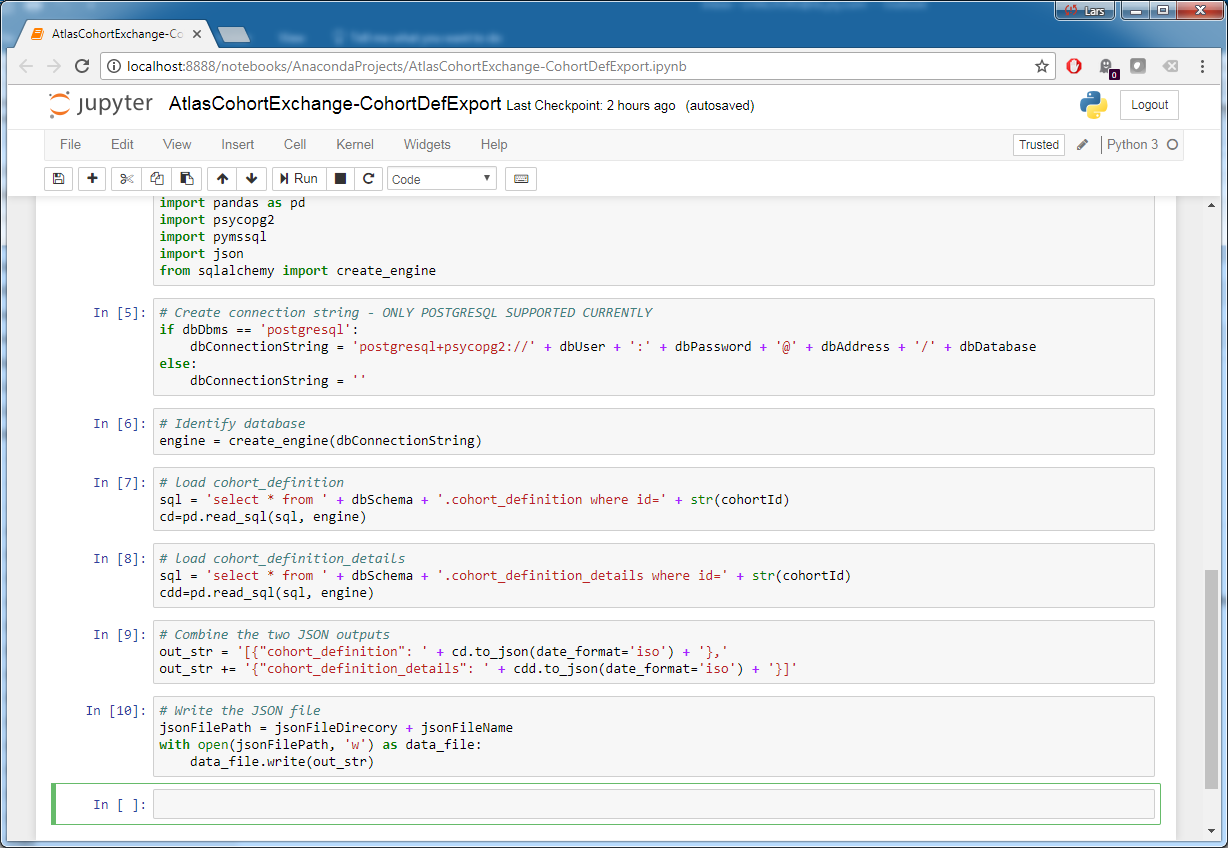
Open the [AtlasCohortExchange-CohortDefExport](https://github.com/solventrix/AtlasCohortExchange-ResultImport/blob/master/AtlasCohortExchange-CohortDefExport.ipynb) Jupyter notebook on a locally accessible Jupyter server:



The content of the Jupyter notebook is rather self-explanatory – modify/update the cell with the database, file path, and cohort ID parameters, and run all the cells in the notebook:



* The database parameters specify the parameters needed to access the ‘webapi’ schema for the central Atlas database.
* The Cohort ID specifies which cohort definition should be exported.
* The directory and file name parameters specify where to save the exported JSON file.



Once all the cells in the Jupyter notebook have been executed, the cohort definition can be found in the JSON file for which the name and path was specified in the parameters section.

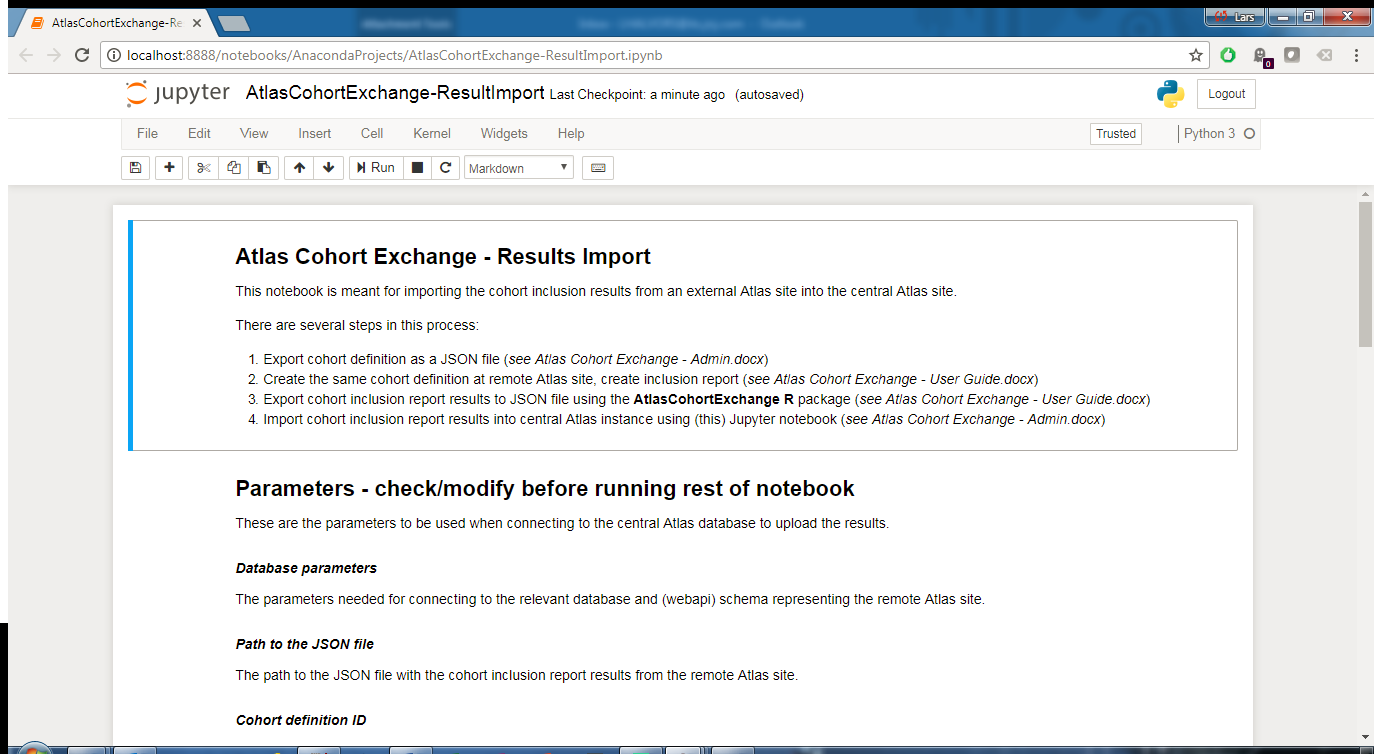
Send the exported JSON file to the external Atlas site(s) with the instructions found in the *Atlas Cohort Exchange – User Guide.docx* document.

# Import cohort inclusion report results from external Atlas site

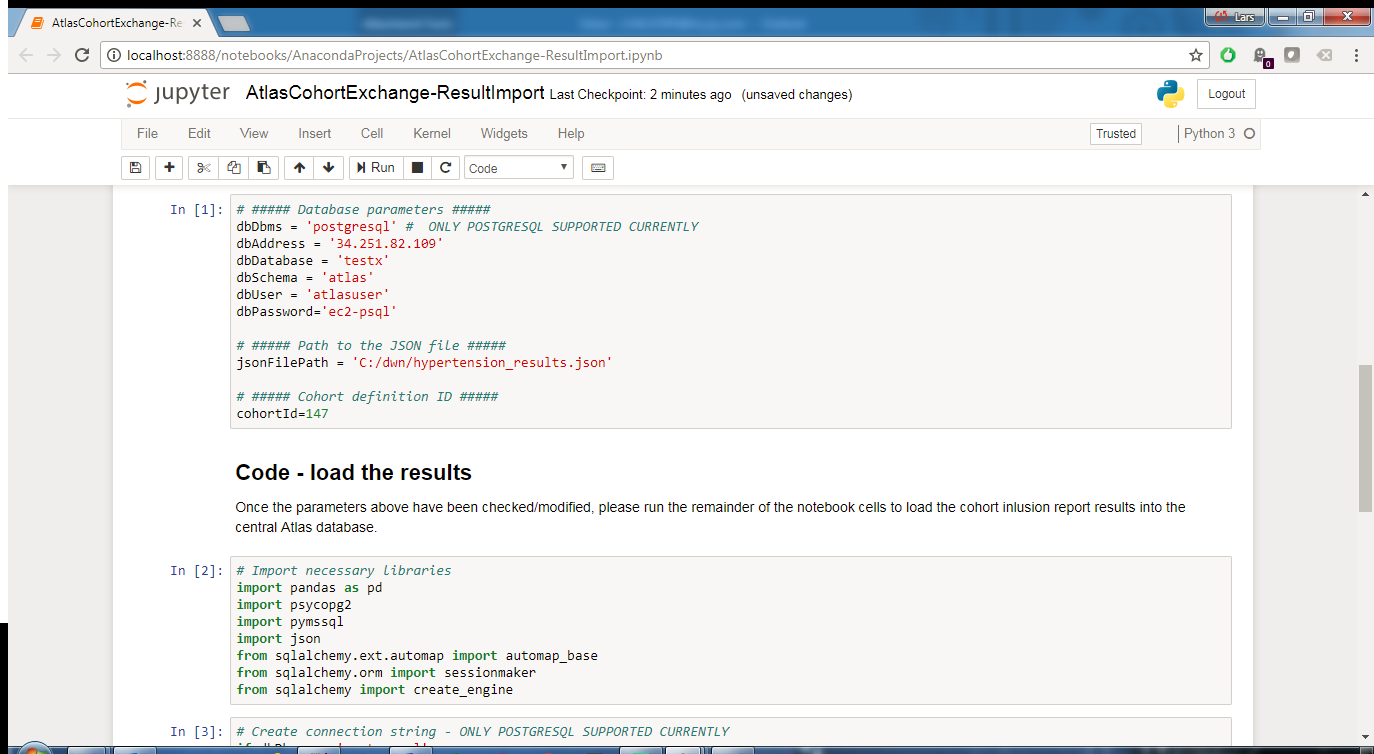
Once the external Atlas site has imported the Cohort definition, generated the local inclusion report, and exported the cohort inclusion report results, these results can be imported back into the central Atlas instance for review.

The cohort inclusion report results from the external Atlas site are sent back to the central Atlas site in a JSON file, which can be imported into the central Atlas instance using the AtlasCohortExchange-ResultImport.ipynb Jupyter notebook.

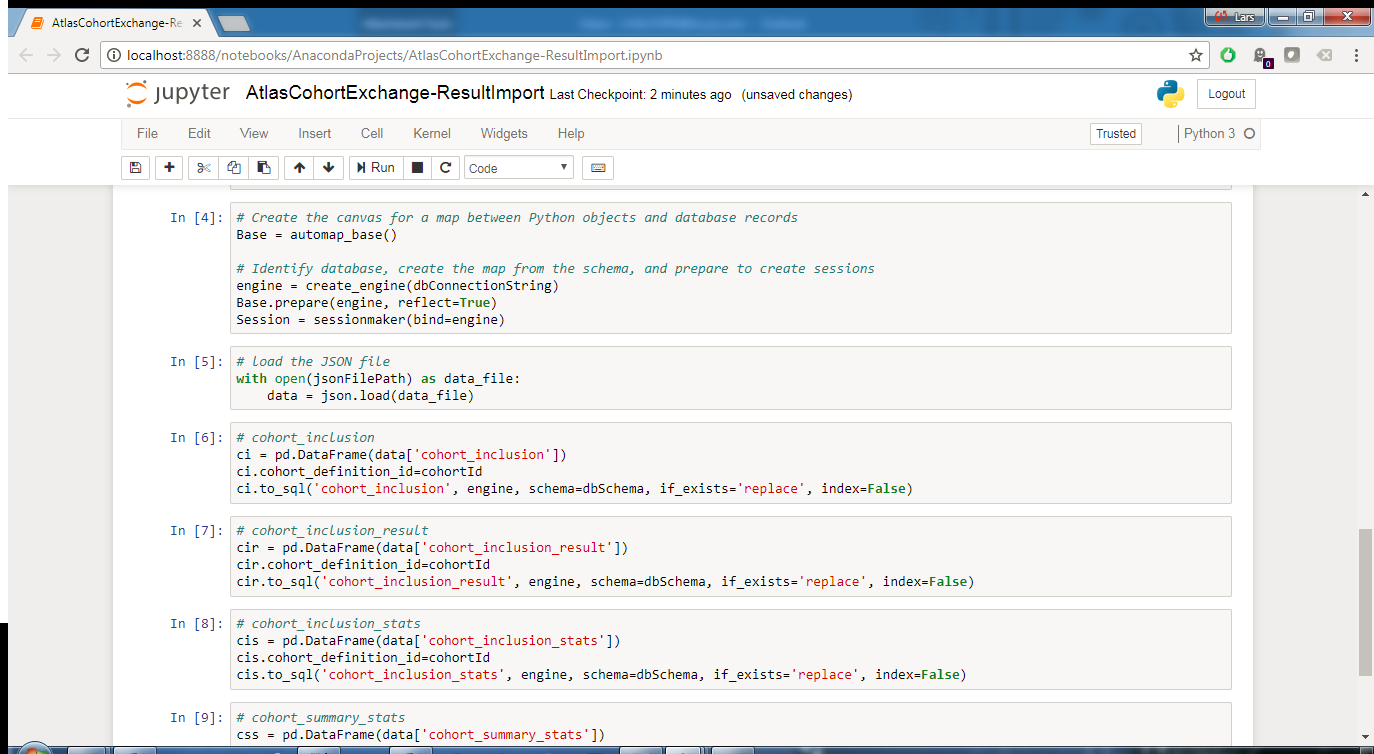
Open the [AtlasCohortExchange-ResultImport](https://github.com/solventrix/AtlasCohortExchange-ResultImport/blob/master/AtlasCohortExchange-ResultImport.ipynb) Jupyter notebook on a locally accessible Jupyter server:



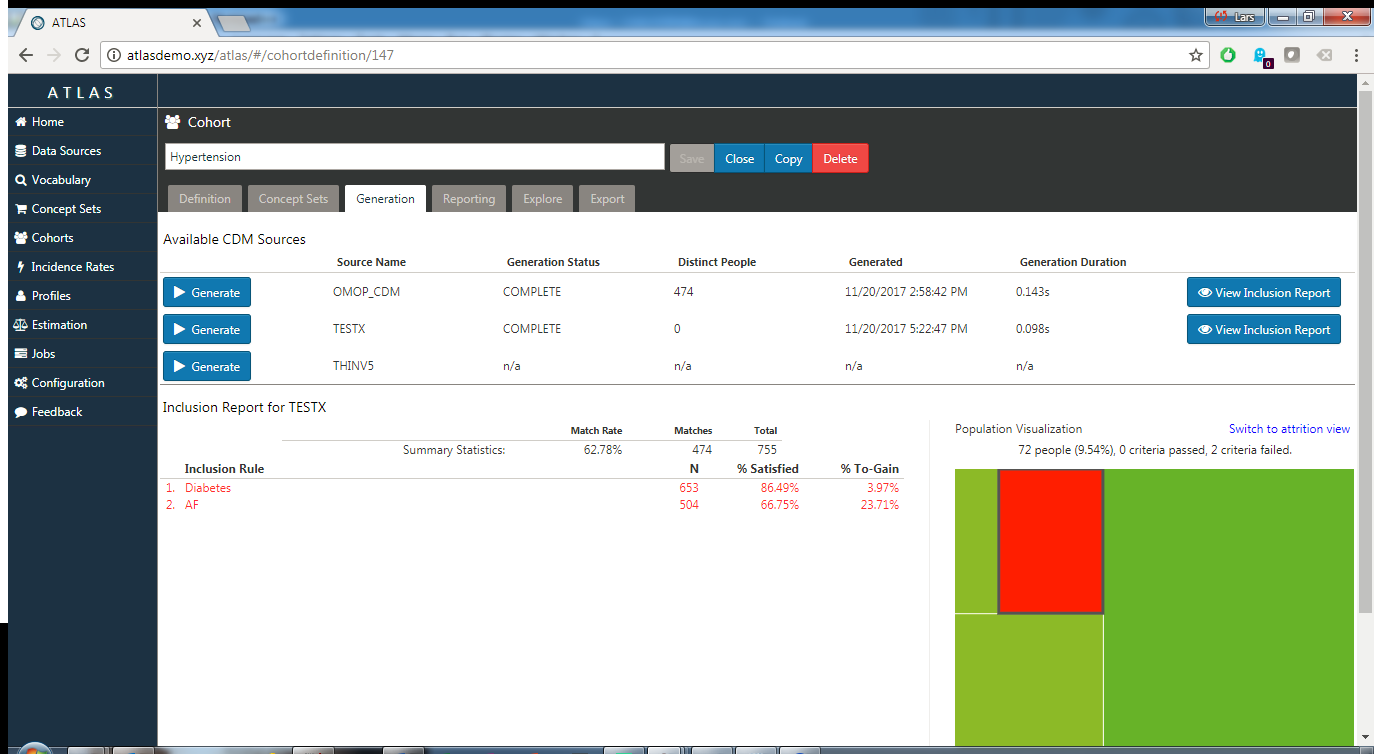
The content of the Jupyter notebook is rather self-explanatory – modify/update the cell with the database, file path, and cohort ID parameters, and run all the cells in the notebook:



* The database parameters specify the parameters needed to access the ‘webapi’ schema for this external Atlas site in the central Atlas database.
* The file path parameter specifies where the JSON file with the cohort inclusion report results for the external Atlas site is located.
* The Cohort ID specifies the ID in the central Atlas instance of the cohort for which the Cohort definition was sent out, and for which the results are now being read back in for this external Atlas site.



Once all the cells in the Jupyter notebook have been executed, the cohort inclusion report results for this external Atlas site can be reviewed in the central Atlas instance:



**NOTE**: Once the cohort inclusion report results for this external Atlas site have been imported, DO NOT click the **Generate** button for this site again – this will clear out the imported results!