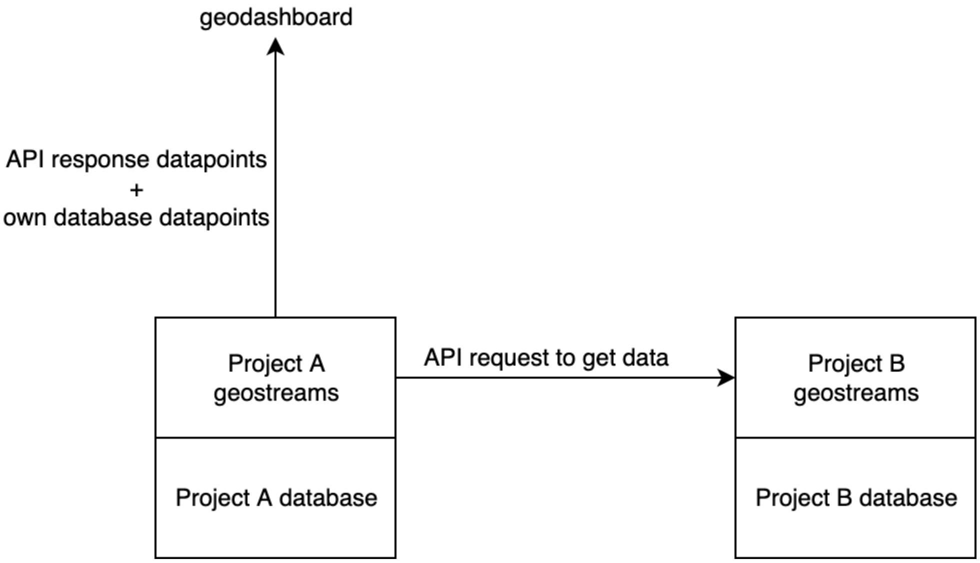
Sharing data between multiple projects

This page is for discussing ways in which we can accommodate multiple projects using same sensors without duplicating data.

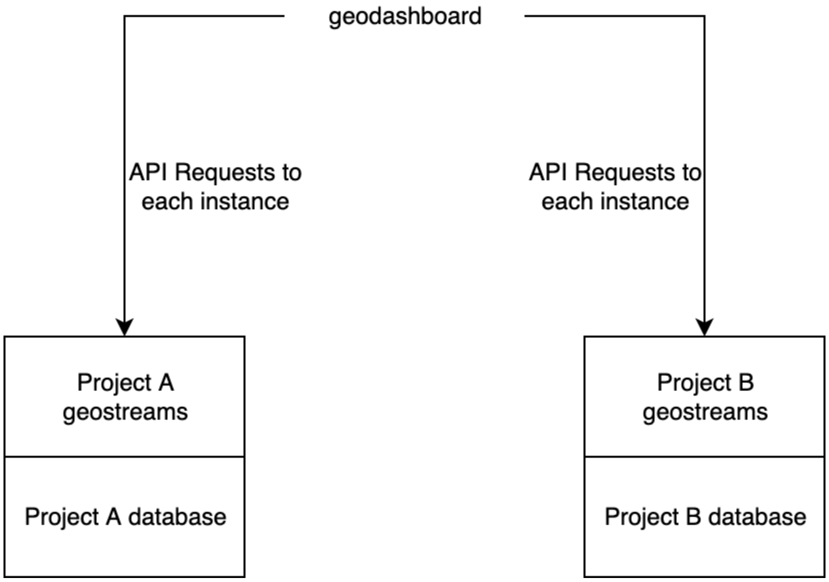
Approach 1- Aggregation and Filtering on backend

- Have the backends make requests between them and filter the data in the backend before returning to geodashboard.
- The api returns aggregated data from own database and data recieved from API requests to other projects.



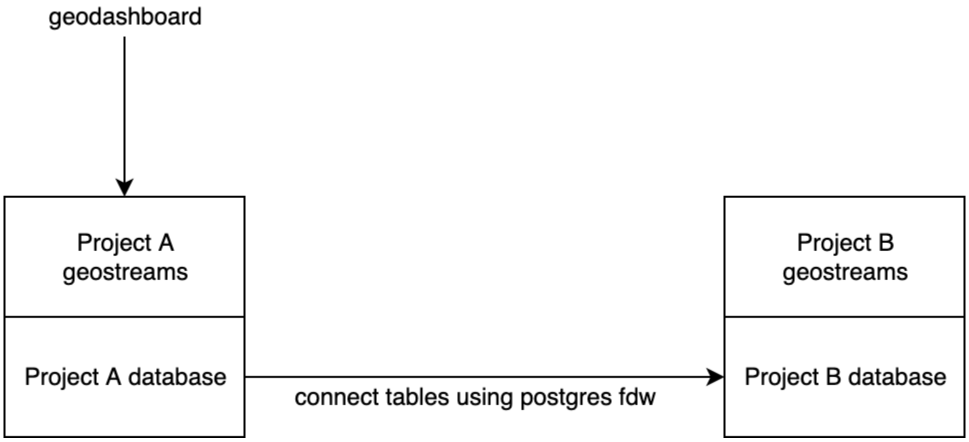
Approach 2 -Aggregation and Filtering on frontend.

The geodashboard communicates with the geostreams-api instances directly. The filtering and aggregation of data takes place in the geodashboard app.



Approach 3- Connecting tables of different projects

Use postgres partitioning and foreign data wrapper to connect tables of services with each other.



How:

Scenario: Project A needs to show all datapoints of a source in Project B.

- Partition the Project B database tables on sources. i.e. datapoints is partitioned into datapoints_sourceX, datapoints_sourceY.
  - Queries can still be run on the main datapoints table returning data for all the sources
- Create a partition table in Project A's database `datapoints_sourceX`.
- Use foreign data wrapper to connect `datapoints_sourceX` partition table to Project B's datapoints_sourceX table.
- Now, all the data of datapoints_sourceX will be available in datapoints table on Project A as well without it being copied into Project A database.

Drawbacks:

- Will require restructuring the database schema.
- The databases on each machine will have to be exposed so that they can communicate. Currently, they are only accessible on the machines locally.

Link for more information:

https://www.percona.com/blog/2019/05/24/an-overview-of-sharding-in-postgresql-and-how-it-relates-to-mongodbs/

https://pgdash.io/blog/postgres-11-sharding.html