Workshop 2: (Near) Real Time Data Analytics and Integration

Introduction:

Cloud native geospatial workflows enable the integration of data from many different sources much more seamless, reducing the need for multiple file formats and ETL processes.

In this workshop we will look at how we can use CARTO + Snowflake to quickly and easily integrate geospatial data from different sources into a real world use case.

This will enable us to investigate geospatial data available through:

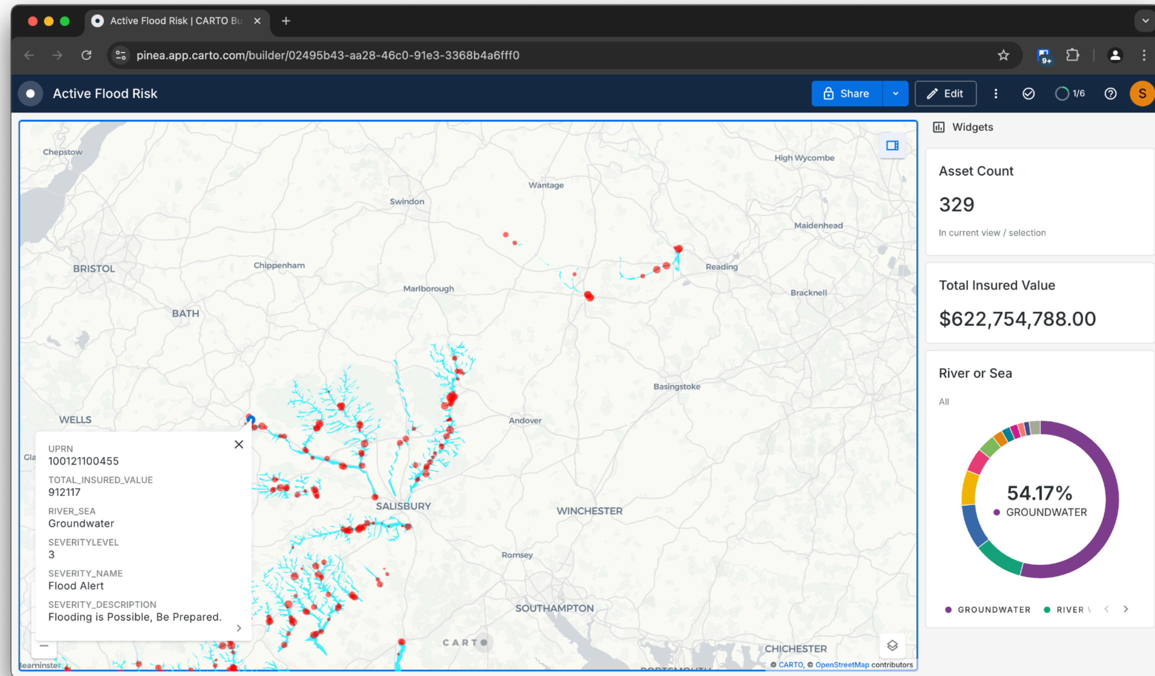
- The CARTO Spatial Data Catalog
- The Snowflake Marketplace
- Live API connections
- External files.

Whilst learning about the following CARTO and Snowflake functionality:

- CARTO Builder Visualisations
- CARTO Workflows
- Snowflake Marketplace (data and apps)
- Snowflake data sharing & Snowpipes

The use case

In Britain we love nothing more than talking about the weather, so we're going to simulate a portfolio of physical asset locations and build a workflow and dashboard to investigate how they are exposed to live flood risk and current weather events.



Workflow 1: Current Flood Exposure

Datasets:

Environment Agency:

- Flood Areas - .json file
- Real Time Flood - API

Snowflake Marketplace / Ordnance Survey:

- Ordnance Survey UPRN Data
- Ordnance Survey Road Data

CARTO Spatial Data Catalog:

- OSM POIs

Step 1: Live Flood Warnings

The Environment Agency have defined 3,418 flood warning areas across England in which they deem there is significant potential for flooding.

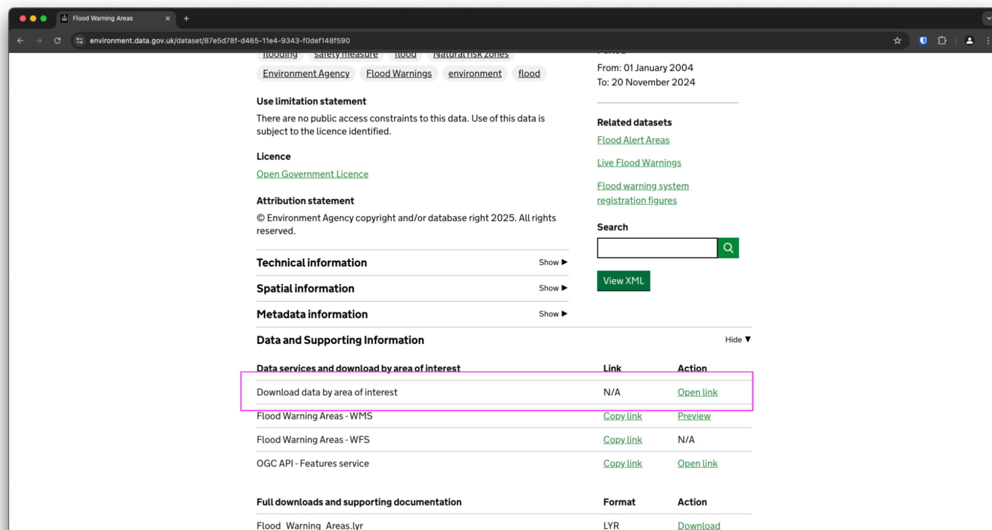
The Real Time Flood API can then be used to surface live flood warning information for these areas alongside the river level and flow information from a network of ~50,000+ gauges across the country.

Flood areas:

The flood areas are obtainable via the API however as the polygons are large (and detailed) it is recommended to store a copy of these locally.

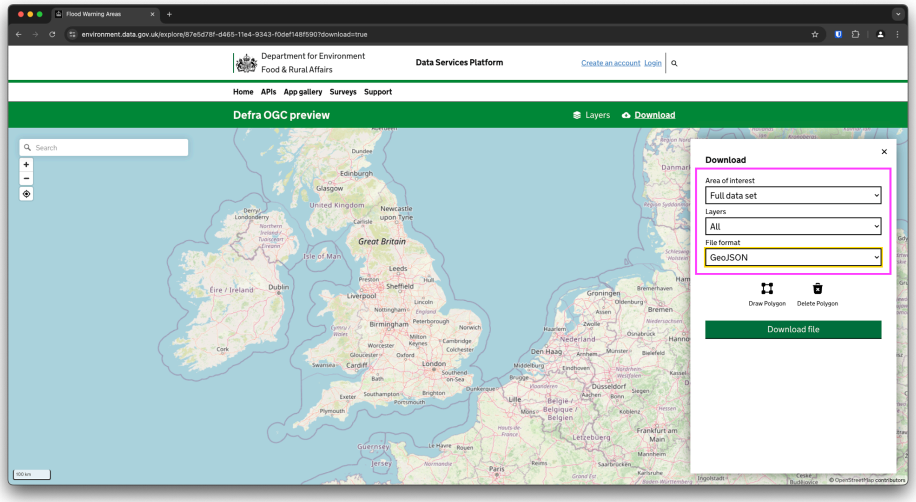
These can be obtained from the DEFRA Data Platform:
https://environment.data.gov.uk/dataset/87e5d78f-d465-11e4-9343-f0def148f590

1. Click the link above and select “Download data by area of interest”

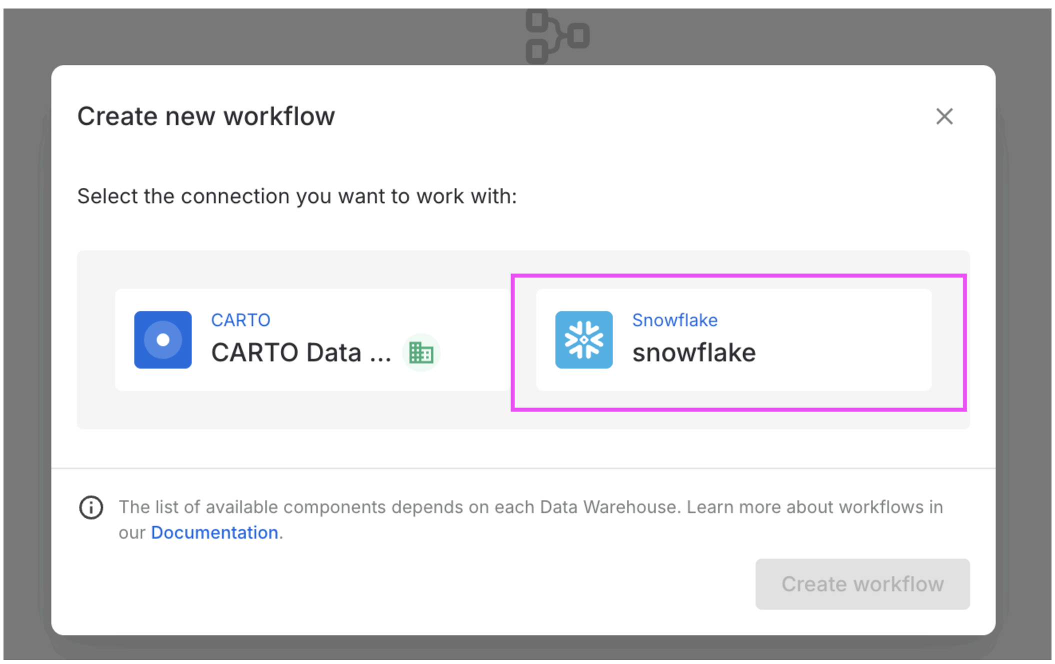


2. On the resultant page, select the following:
    a. Area of interest: Full data set
    b. Layers: All

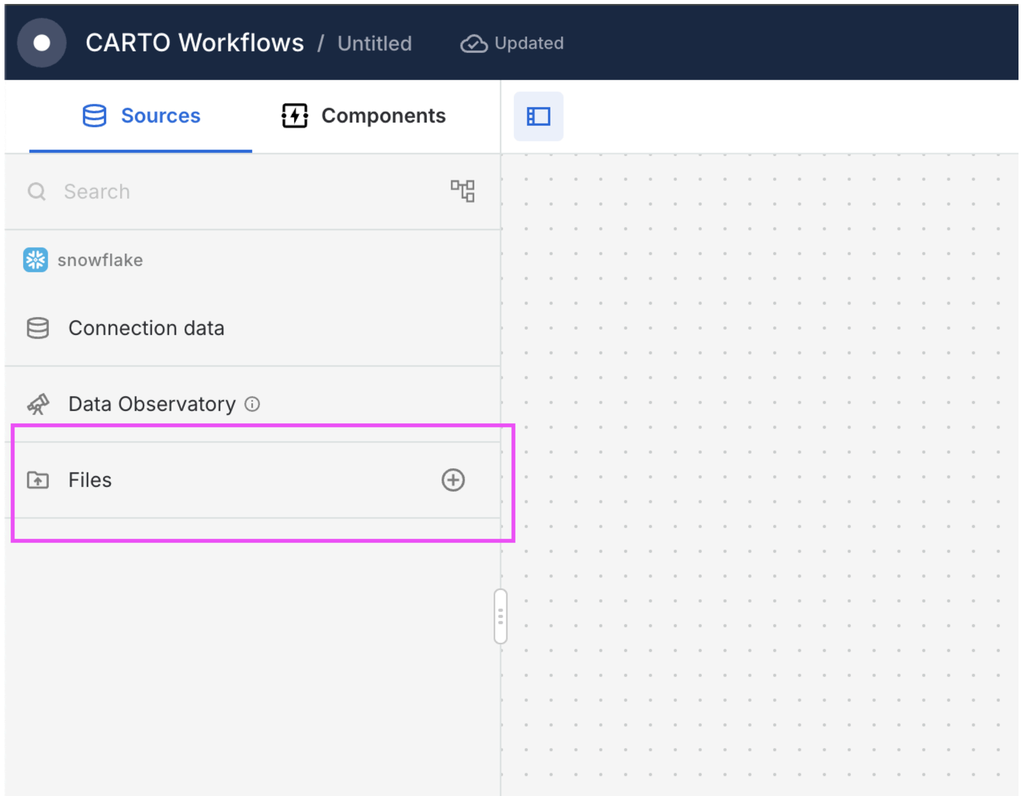
c. File format: GeoJSON



3. Next, we are going to head to CARTO and create our first Workflow. Go to the Workflows tab on the left and select **New Workflow**. Make sure to select your Snowflake connection on the next screen:



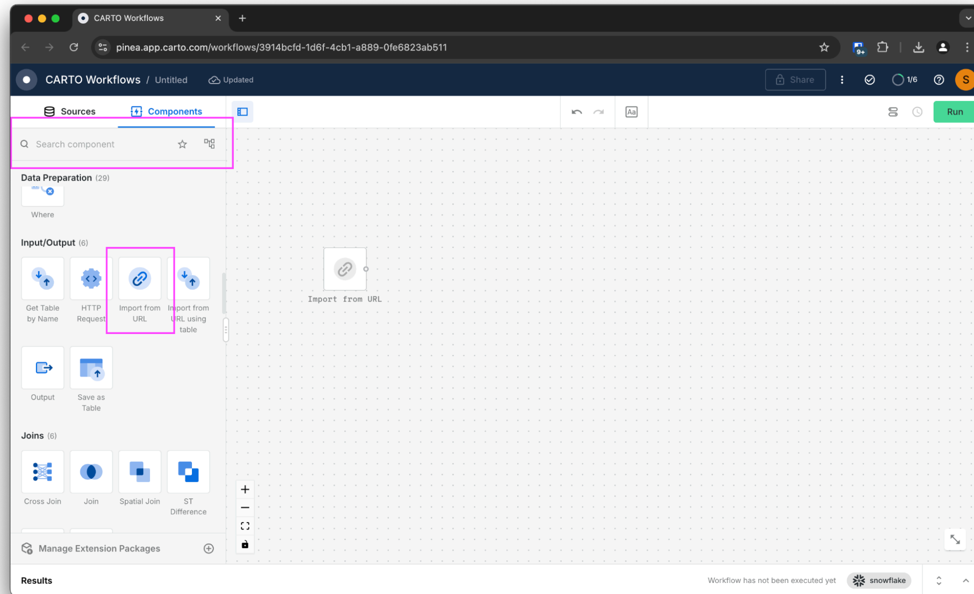
4. In the sources tab, click the + next to files to upload the flood warning areas to temporary storage for this workflow:



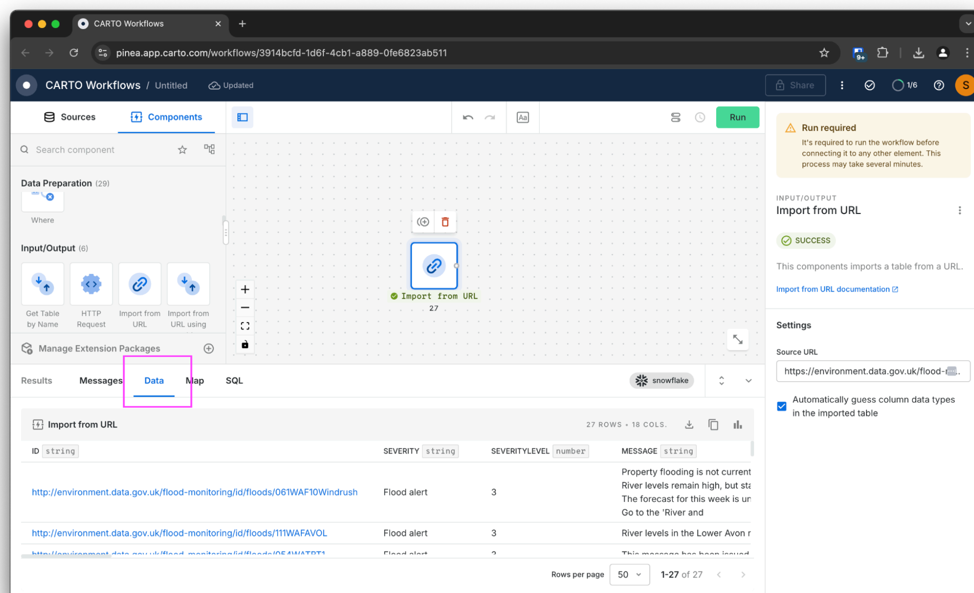
5. Whilst waiting for this to upload, we can set up the **Import from URL component** to obtain the current flood warnings from the Environment agency API. Navigate to https://environment.data.gov.uk/flood-monitoring/doc/reference#flood-warnings where we can see that the current floods can be obtained from https://environment.data.gov.uk/flood-monitoring/id/floods. Click on this link to view the data that is exposed by the API.

6. This endpoint lets you retrieve data in different formats so lets append .csv to the end of the URL to obtain the data in CSV format for easy import. https://environment.data.gov.uk/flood-monitoring/id/floods.csv
7. Back in CARTO Workflows, go to the components panel and find the **Import from URL** component. You can also search for it. Drag it onto the workflow

canvas:

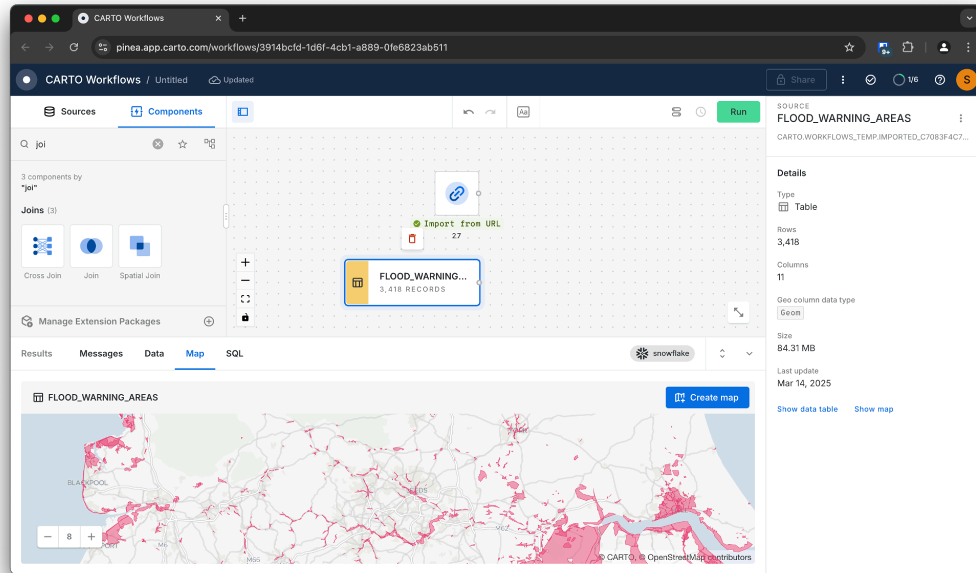


8. Click on it and paste the URL in the UI, then run the workflow. Once it has run, click on the Data tab to preview the table of API results showing live flood warnings:



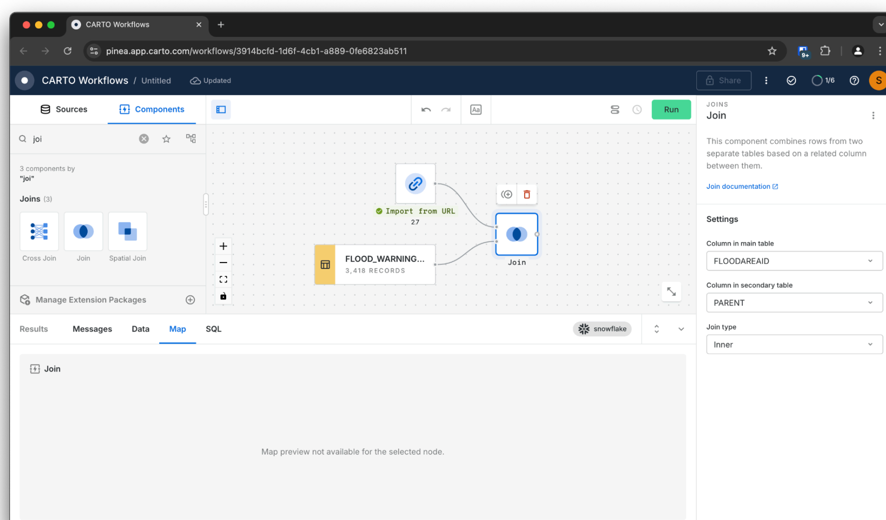
9. To visualise these, we can now join the Live API flood warnings to the flood warning area polygons that we set to upload. Go back to the Sources tab > Files and drag the flood warning areas that we previously imported into the workflow canvas.

Click on the Flood Warning areas to get a Data and Map preview:

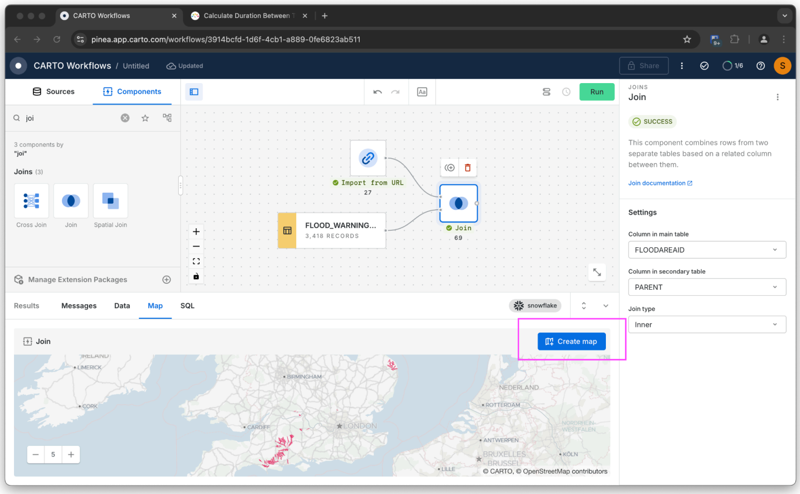


10. Next add a Join component. We are going to join on:
    a. FLOODAREAID - from the API table
    b. PARENT - from the Flood Warning Areas table

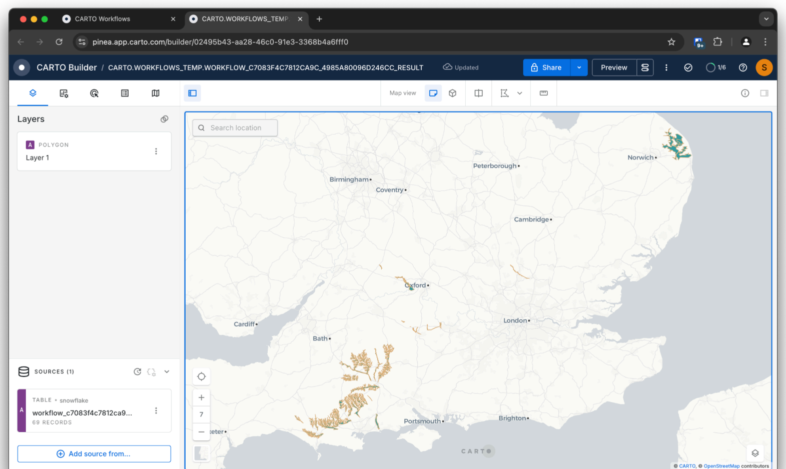
    Here we are going to choose an INNER join type so we only map flood warning areas which currently have an active flood warning rather than all 3000+ areas.



11. Run the workflow and click on the map preview of the Join component, then select the blue button: Create Map.



This will open CARTO Builder, a drag and drop map building interface, pre-populated with current flood warning areas:



Based on what you learnt in Helen’s workshop, have a go at styling the active flood warning areas by severity.

Recap:

So far via CARTO we have loaded in a GeoJSON file of potential flood warning areas, called the

Environment Agency Flood API to retrieve active flood warnings and joined these in a temporary Snowflake table.

Connecting to API's directly in Snowflake.

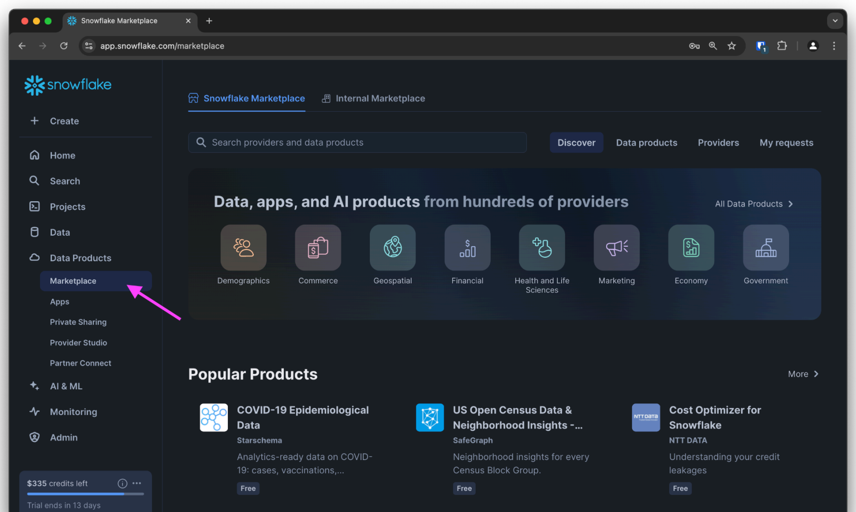
Step 2: Testing portfolio exposure.

Now we have active flood warning areas, the next step is to investigate the potential impact on our asset portfolio.

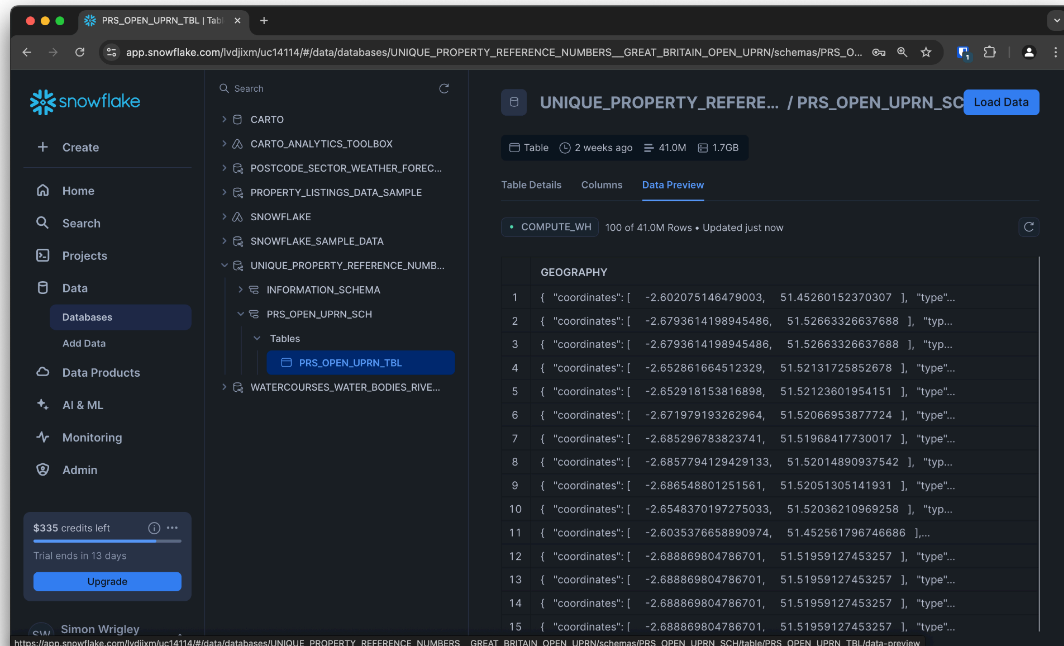
To simulate an insurance policy / commercial asset portfolio, we are going to use the Ordnance Survey UPRN dataset which has a centroid point and unique ID for every building in the UK (all 40+ million of them).

As of today, through a new partnership this is available for instant access through the Snowflake Marketplace.

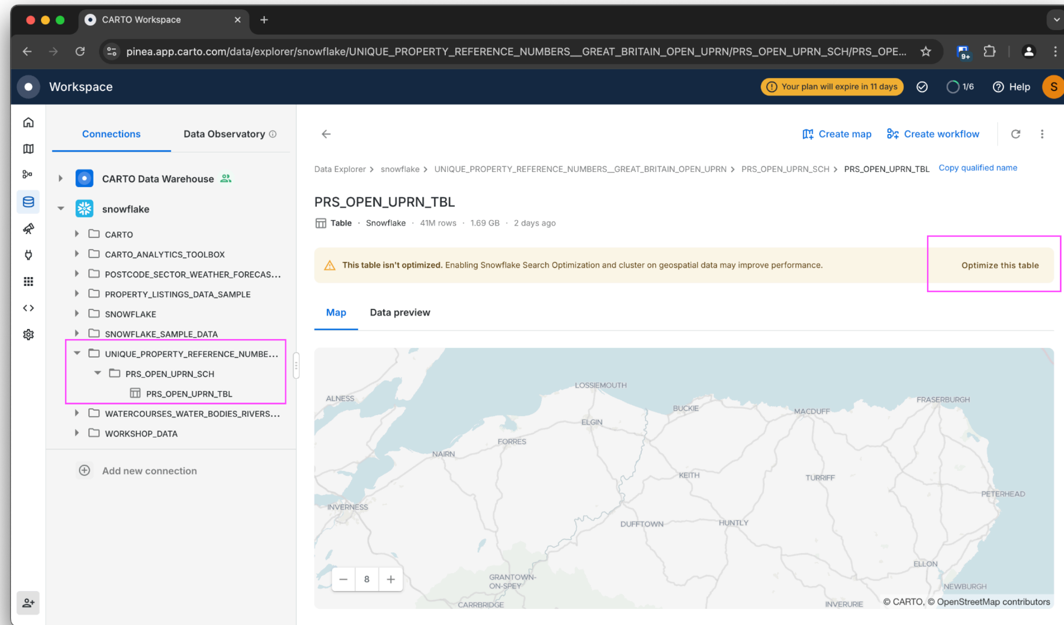
1. Go to your Snowflake trial account and navigate in the left panel to **Data Products > Marketplace.**



2. Search for **Ordnance Survey Unique Property Reference Numbers**. Open the listing and click Get.
3. Once you are subscribed, navigate to **Data > Databases** and you'll find you have a new database: **UNIQUE_PROPERTY_REFERENCE_NUMBERS__GREAT_BRITAIN_OPEN_UPRN** in which you'll find a schema named **PRS_OPEN_UPRN_SCH** in which is a table: **PRS_OPEN_UPRN_TBL** which contains a Geography column, the UPRN and it's coordinates.



4. Whilst we could use this data immediately, it is read only and we're going to modify it slightly. First, lets create new Database to save our outputs in:
    a. In Snowflake, click on **Data** on the left and then create a new database using the blue **+ Database** button in the top right.
    b. Name it **workshop_data**
    c. Click on the newly created database and add a new schema called **UPRN**.
5. Back in CARTO, we are going to create a geospatially optimised version of the UPRN table. Navigate to the Data Explorer and find the Ordnance Survey UPRN dataset. Click Optimize Table in the yellow banner:



In the popup, select the **workshop_data** database and **UPRN** schema that we just created in snowflake. We can save the table as UPRN_optimized.

This performs two Snowflake operations under the hood as it creates a new table:

a. Orders the table by the ST_GEOHASH of the geometry column.
b. Enables Snowflake Search Optimisation.

6. To enable us to add further financial context to our end dashboard, lets add a column to this dataset called **total_insured_value** and populate it with some fabricated data.
   a. In Snowflake, create a new SQL Worksheet (using the + button on the left)
   b. At the top of the Worksheet, set the worksheet context to your **workshop_data** database and **UPRN schema**

   Copy and paste the following code, making sure the table name matches your optimized URPN table created above

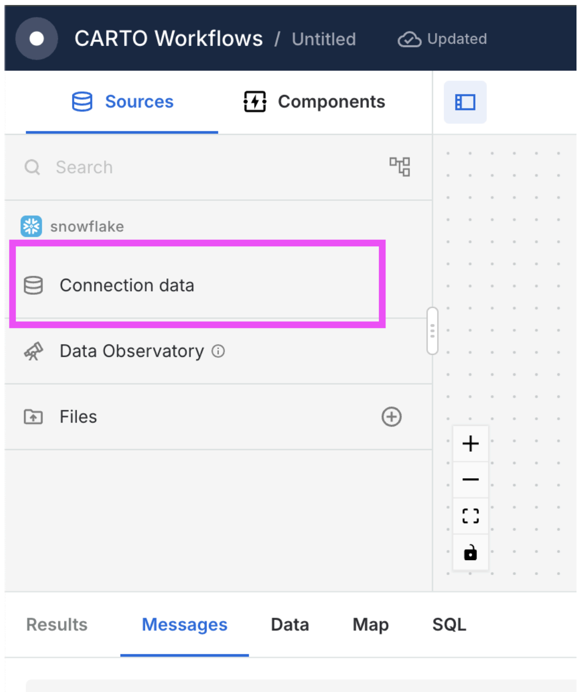
Unset

```
ALTER TABLE UPRN_optimized ADD COLUMN total_insured_value float;

UPDATE UPRN_optimized SET total_insured_value = UNIFORM(100000 ,
    2000000 , RANDOM() );
```

Here we are creating a uniform distribution of values between 100k and 2m using Random Numbers to simulate the total insured value for a portfolio of assets.

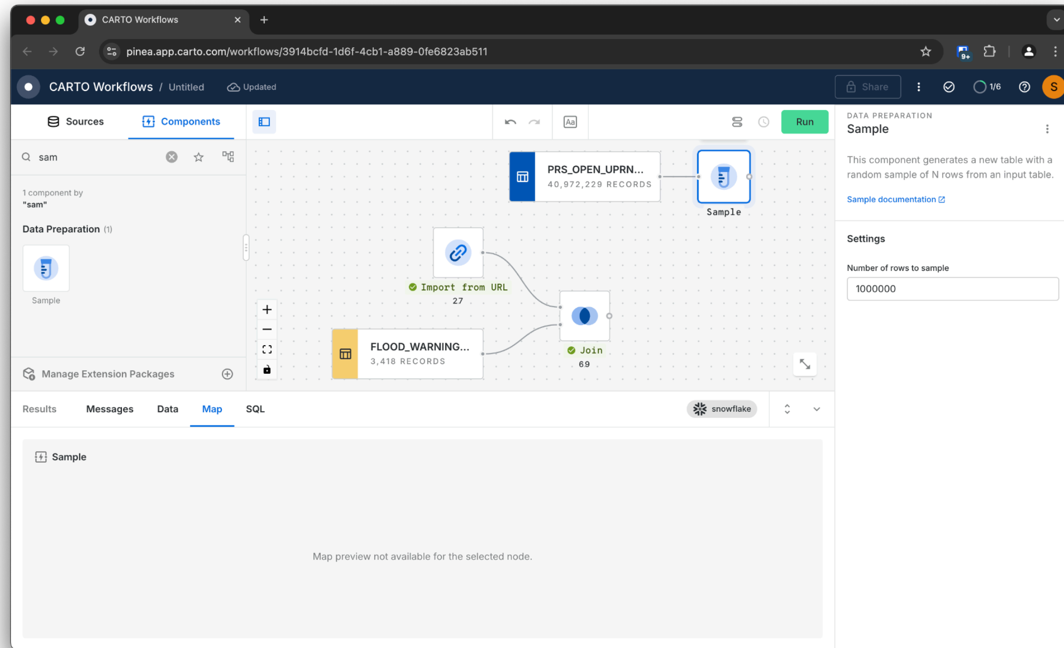
7. Now that this is available in your Snowflake account, it's also automatically available for use in CARTO through the direct connection. Go back to your Workflow and open the Sources tab - Select **Connection data**



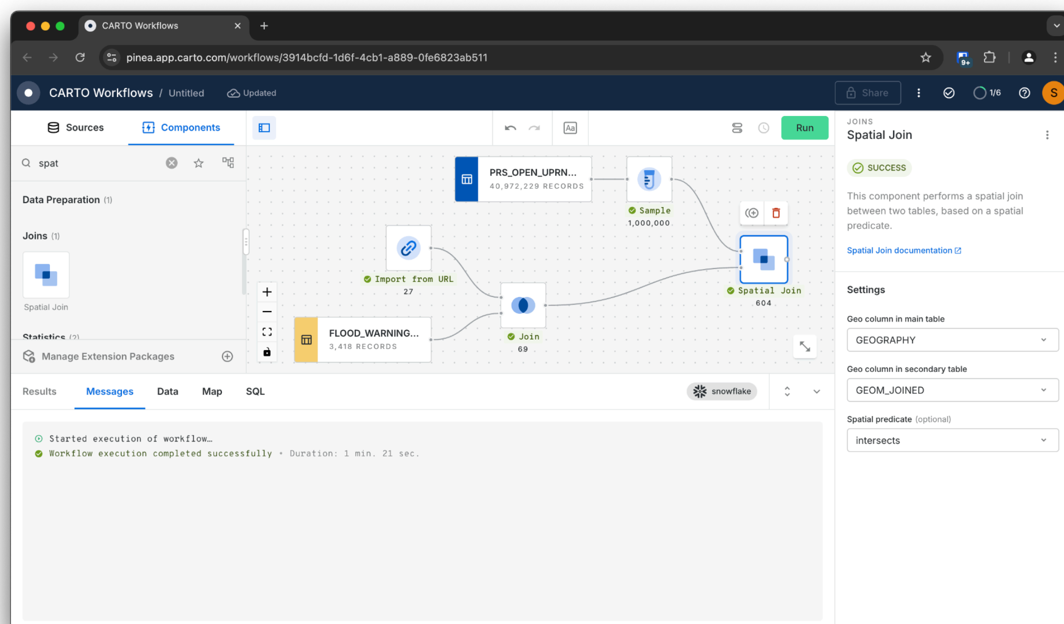
Here you should recognise the contents as it is the same as what we just saw in the Snowflake trial. Navigate through and drag the **UPRN_optimized** table onto your Workflow Canvas.

8. As we are not going to be insuring every single building in the UK, drag on the **Sample** component to select a random subsample of the UPRN points to simulate our hypothetical asset portfolio. Select a reasonable number for testing e.g.

1,000,000 (this number can also be adjusted later).



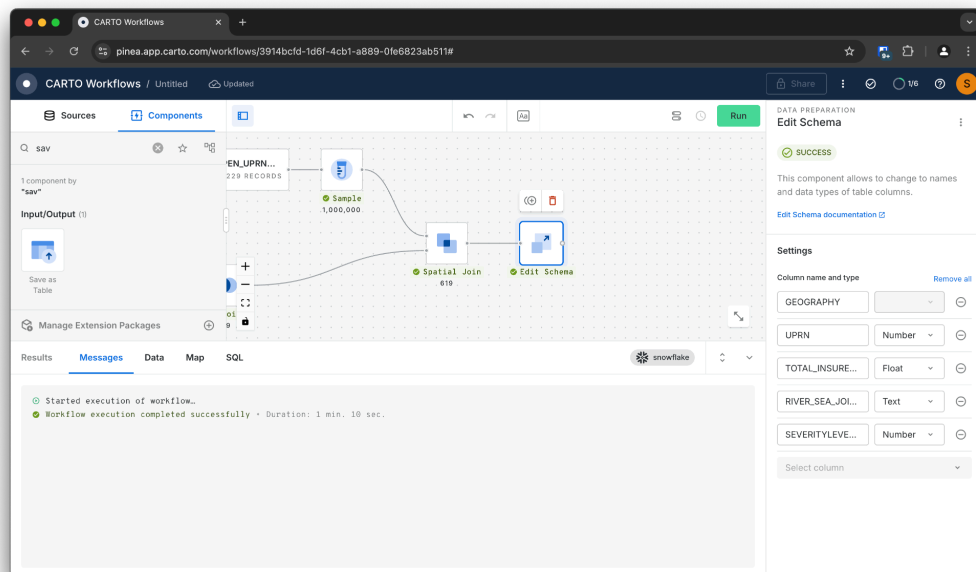
9. Next add a **Spatial Join** component. Here we need to select the two geom / geography columns and the spatial join type:



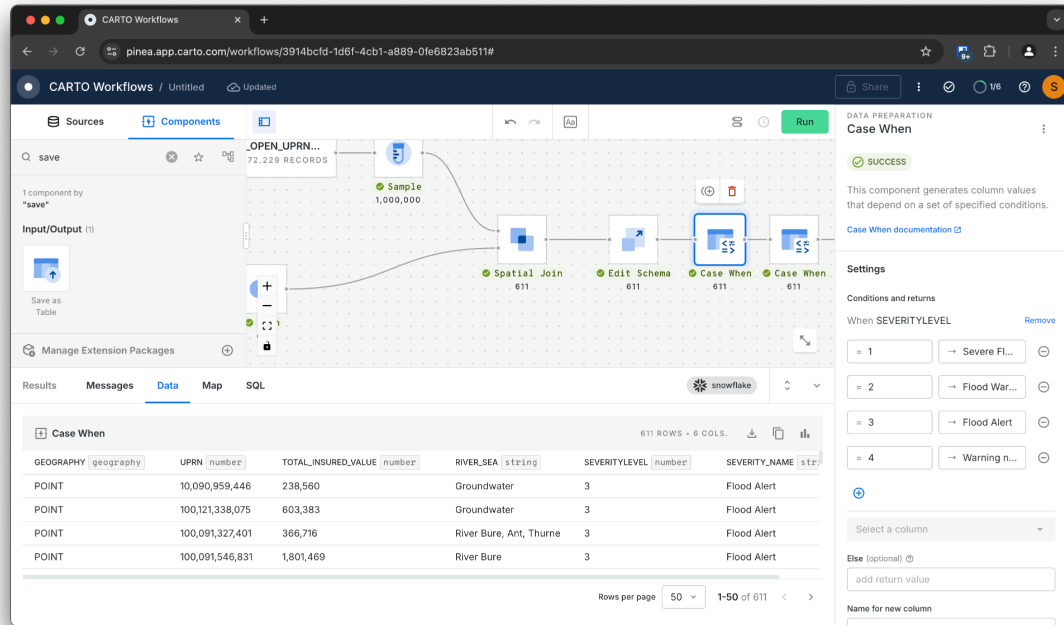
If your workflow is setup the same as in the screenshot with the UPRN dataset as the **main** table (top node) and active flood warnings as the **secondary** table (bottom node):

   a. **Intersects**: will give you only the buildings in an active flood warning area
   b. **Left Join**: will give you all buildings, regardless of whether they are in a flood zone or not.

10. Prior to saving the table we are going to use the **Edit Schema** component to select and rename only the columns we need.
    a. Drag the component into the workflow canvas
    b. Add the following fields:
       i. GEOGRAPHY
       ii. UPRN
       iii. TOTAL_INSURED_VALUE
       iv. RIVER_SEA_JOINED_JOINED
       v. SEVERITYLEVEL_JOINED
    c. Remove the _JOINED suffixes to give more sensible column names (don't use spaces).



11. For our final dashboard we are going to add in a **CASE When** component to add descriptions of the flood severity levels.

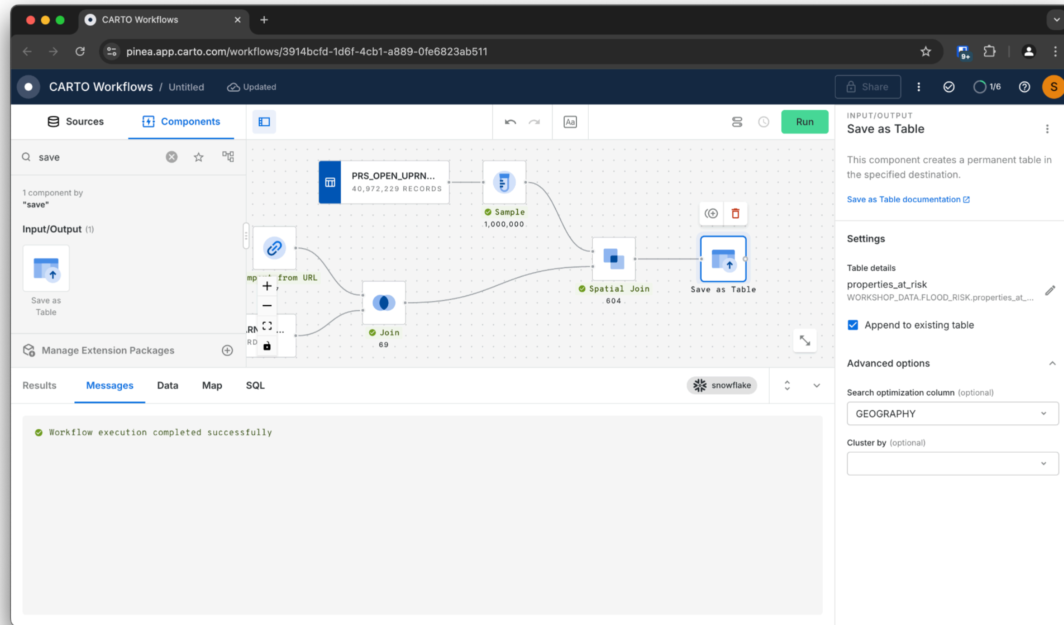


Choose the SEVERITYLEVEL column and add conditions using the following name column:

Level	Name	Description
1	Severe Flood Warning	Severe Flooding, Danger to Life.
2	Flood Warning	Flooding is Expected, Immediate Action Required.
3	Flood Alert	Flooding is Possible, Be Prepared.
4	Warning no Longer in Force	The warning is no longer in force

Optional: Add a second CASE WHEN component to add in the description.

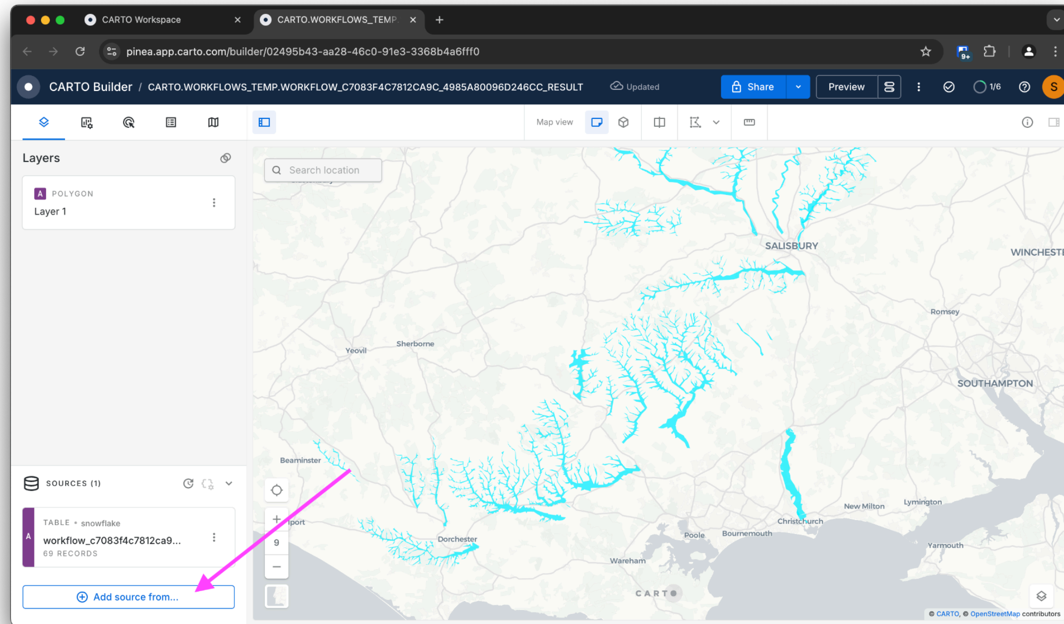
12. Finally add a **Save as Table** component. Choose the **worshop_data** database and the **UPRN** schema that we created earlier and name this new table **properties_at_risk**



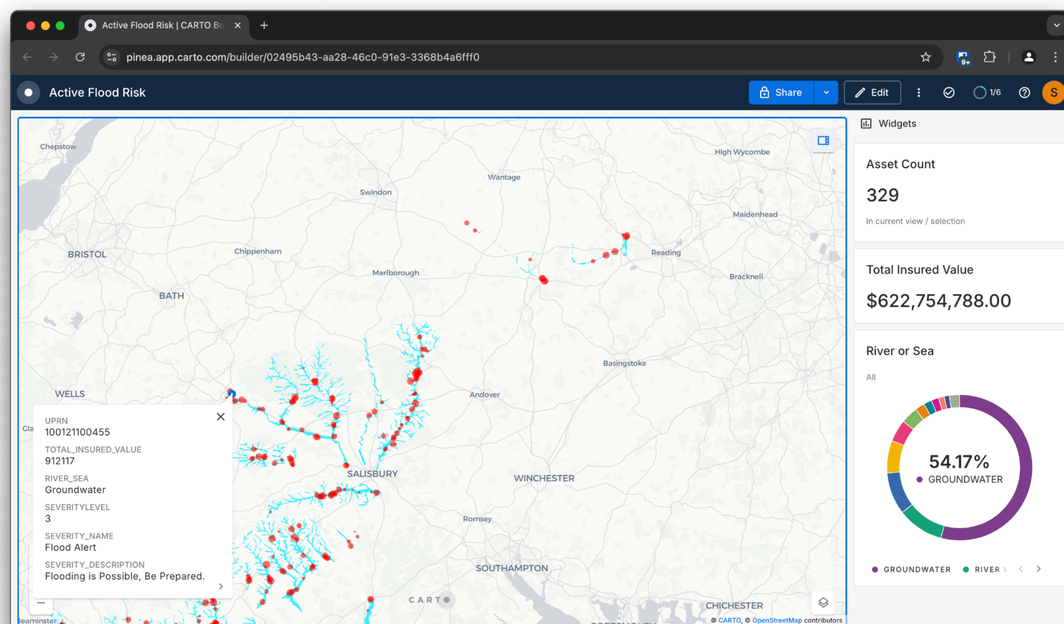
Selecting **Append to existing table** means that every time we re-run this workflow, results will be appended rather than overwritten allowing us to view exposure over time. Under **Advanced options**, selecting the **Geography** column as the Search Optimisation column will ensure efficient reading / visualisation of the data.

13. Run the workflow!
14. If you still have the Flood area map created in [step 11](#) open, navigate back to that tab or alternatively, in the main CARTO window, find it in the maps tab.

15. Click **Add Source From** and add your newly created table to the map.



16. Experiment with adding widgets based on the property layer and hover over interactions from the flood layer. You can rename the layers and style them to create an informative dashboard of current asset flood risk.

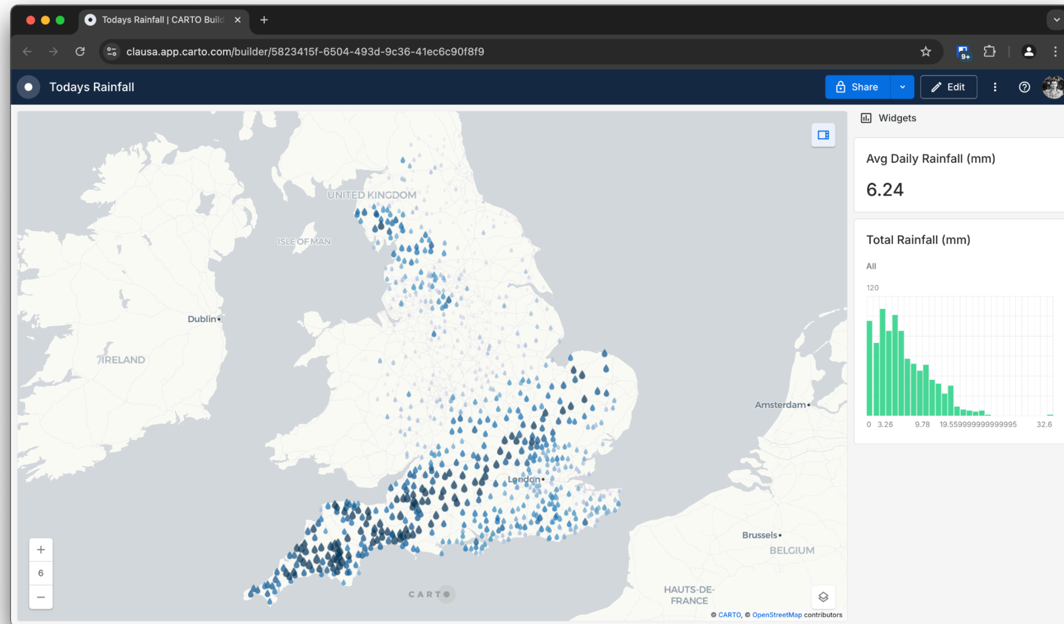


17. Give your map a name and using the share button, feel free to publish it and send a link to your colleagues!

Extension Exercises

If you have got through the material above, please feel free to use the time to try some of these suggested extensions:

1. Integrate data about the daily rainfall amount.



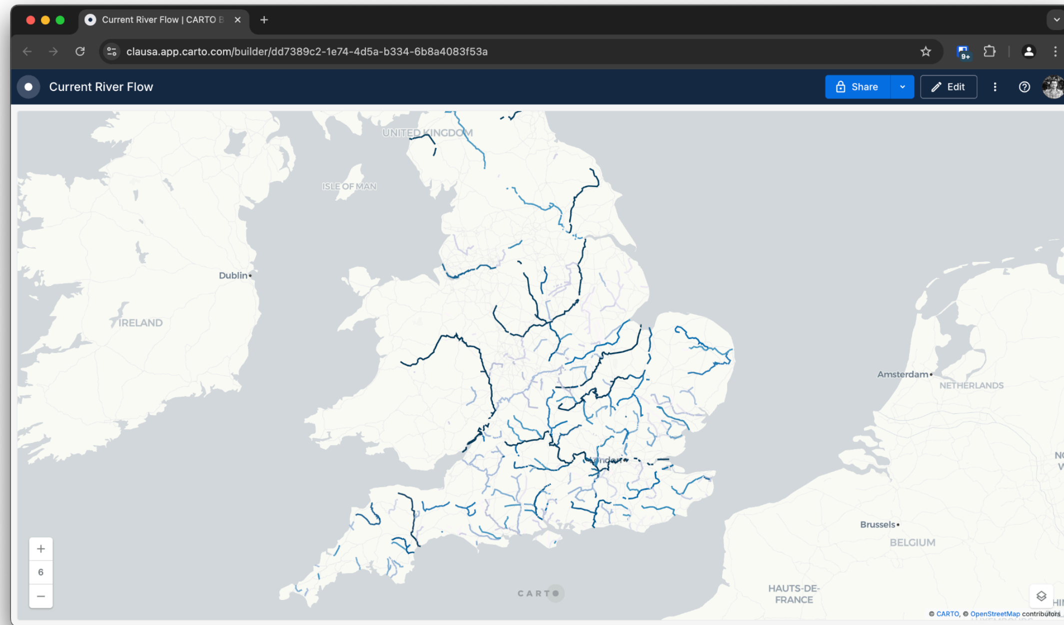
Hints:

a. You can obtain the river monitoring stations which have a rainfall gauge using the following API endpoint: https://environment.data.gov.uk/flood-monitoring/id/stations.csv?parameter=rainfall

b. You can obtain the rainfall readings for today using: https://environment.data.gov.uk/flood-monitoring/data/readings.csv?parameter=rainfall&today&_view=full

c. You might want to use a **Group By** component to aggregate the rainfall readings per station to get a daily total.

2. Combine the Ordnance Survey River dataset (available on the Snowflake marketplace) with live river flow data from the Environment Agency API:



Hints:

a. You'll notice that the Ordance Survey river dataset is split at every branch or junction of a river resulting in a dataset of 192.4k rows. As a first step, you might want to create a new rivers table, joining the geometry segments based on their start and end nodes, grouping by the river name. This can be achieved with the following:

Unset

```
SELECT a.watercourse_name, ST_UNION_AGG(a.geography) as geom
FROM PRS_WATERCOURSE_LINK_TBL a
JOIN PRS_WATERCOURSE_LINK_TBL b
ON a.end_node = b.start_node
AND a.watercourse_name = b.watercourse_name
GROUP BY a.watercourse_name
```

b. Monitoring station locations that measure river flow can be accessed here: https://environment.data.gov.uk/flood-monitoring/id/stations.csv?parameter=flow

c. Current flow readings can be obtained here: https://environment.data.gov.uk/flood-monitoring/data/readings.csv?param eter=flow&_view=full&latest