How to Build a BI Dashboard Using Google Data Studio and BigQuery

Overview

For as long as business intelligence (BI) has been around, visualization tools have played an important role in helping analysts and decision-makers quickly get insights from data.

In this lab you'll learn how to build a BI dashboard with Data Studio as the front end, powered by BigQuery on the back end. It assumes some familiarity with those products. For more information, review the background docs (BigQuery concepts, Data Studio overview).
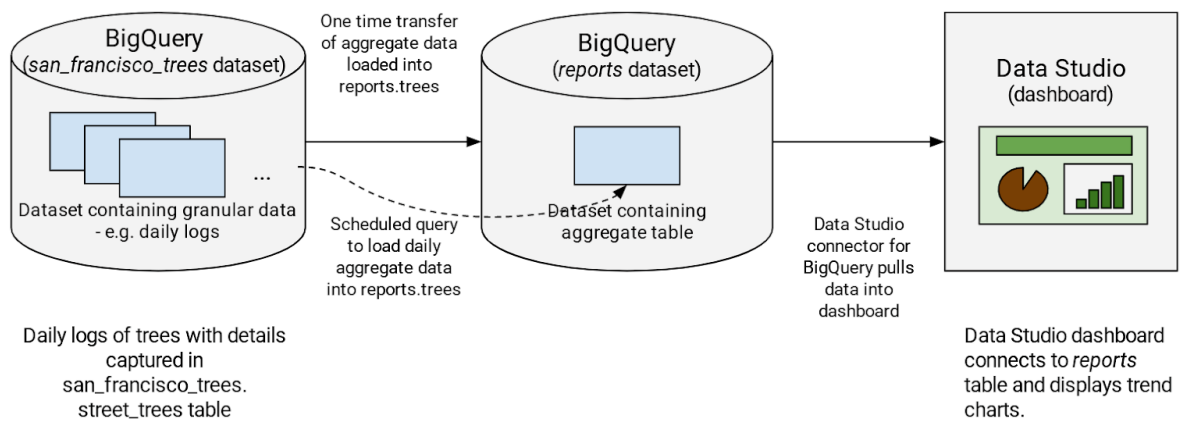
Usecase

For this lab you'll be a manager of tree services for a large city. You make important decisions based on usage logs data, stored in large (multiple TBs) date-partitioned tables in a BigQuery dataset called "Trees".

To get business value out of that data as quickly as possible, build a dashboard for analysts that provides visualizations of trends and patterns in your data.

Solution overview

Typically, a dashboard shows an aggregated view of usage — it doesn't need details all the way to the level of an order ID, for instance. So, to reduce query costs, you'll first aggregate your needed logs into another dataset called "Reports" then create a table of aggregated data. You'll query the table from the Data Studio dashboard. This way, when your dashboard is refreshed, the reporting dataset queries process less data. Since usage logs from the past never change, you'll only refresh new usage data into the Reports dataset.

One time transfer of aggregate data loaded into reports.trees

Scheduled query to load daily aggregate data into reports.trees

Data Studio connector for BigQuery pulls data into dashboard

Daily logs of trees with details captured in san_francisco_trees. street_trees table

Data Studio dashboard connects to *reports* table and displays trend charts.

Uploading queryable data

In this section, you pull in some public data so you can practice running SQL commands in BigQuery.

Open BigQuery Console
- In the Google Cloud Console, select **Navigation menu** > **BigQuery.**
- Click on the **+ ADD DATA** link, then select **Explore public datasets**:
- Search for "trees" and press **Enter**.
- Click on the **Street Trees** tile, then click **View Dataset**.
- A new tab opens, a new project called **bigquery-public-data** is added to the Explorer panel

Create a reports dataset in BigQuery

Next you'll create a new dataset called Reports in your project. A separate dataset has a couple of benefits: it reduces the amount of data queried by the dashboard, and it removes unnecessary access to your source datasets by users who are only interested in aggregated data.

Click on the project name that starts with "Qwiklabs", then click on **View Actions**. Click on **Create dataset** and call it "Reports".

Query the dashboard data

Next you run a one-time query to pull the data for the last year, summarizing:

- The number of trees planted each month
- Which species of trees were planted
- Who the caretaker of the trees is
- Address of the planted trees
- Tree site information

Add the following to the query editor:

```
SELECT
 TIMESTAMP_TRUNC(plant_date, MONTH) as plant_month,
  COUNT(tree_id) AS total_trees,
  species,
  care_taker,
  address,
  site_info
FROM

`bigquery-public-data.san_francisco_trees.street_trees`
WHERE
  address IS NOT NULL
  AND plant_date >= TIMESTAMP_SUB(CURRENT_TIMESTAMP(),
INTERVAL 365 DAY)
  AND plant_date < TIMESTAMP_TRUNC(CURRENT_TIMESTAMP(),
DAY)
GROUP BY
  plant_month,
  species,
  care_taker,
  address,
  site_info
```

Click the **More** button, and select **Query settings** from the dropdown menu.
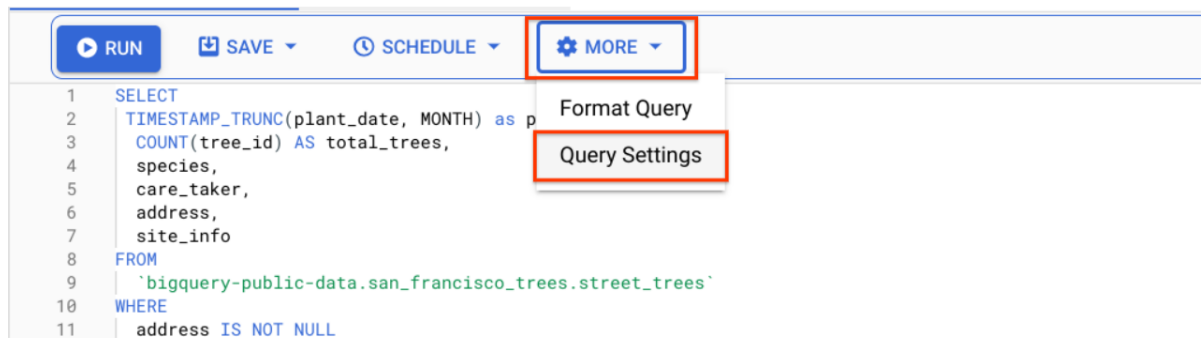
- Your Project name and Dataset name automatically fills in.

- Select **Set a destination table for query results**.

- Create a name for the table, like "Trees".

- For **Destination table write preference**, select **Write if empty**.

Because you specified a **Table name** and selected the **Write if empty** preference, the query creates a table if the table does not already exist.

Accept the other default settings and click **Save**.

Click **Run** to run the query.

When the query completes. You are on the **Results** tab, where you can see the data.

Row	plant_month	total_trees	species	care_taker	address	site_info
1	2021-08-01 00:00:00 UTC	1	Quercus tomentella :: Island oak	Private	247 Naples St	Sidewalk: Curb side : Cutout
2	2021-08-01 00:00:00 UTC	1	Quercus tomentella :: Island oak	Private	242 Naples St	Sidewalk: Curb side : Cutout
3	2021-08-01 00:00:00 UTC	1	Afrocarpus gracilior :: Fern Pine	Private	35 Naples St	Sidewalk: Curb side : Cutout
4	2021-08-01 00:00:00 UTC	1	Ficus retusa nitida :: Banyan Fig	Private	470 Fillmore St	Sidewalk: Curb side : Cutout
5	2021-08-01 00:00:00 UTC	1	Pyrus kawakamii :: Evergreen Pear	Private	2300 17th St	Sidewalk: Curb side : Cutout
6	2021-08-01 00:00:00 UTC	1	Jacaranda mimosifolia :: Jacaranda	Private	1072 Munich St	Sidewalk: Curb side : Cutout

Scheduling queries in BigQuery

To keep your dashboard up-to-date, you can schedule queries to run on a recurring basis. Scheduled queries must be written in standard SQL, which can include Data Definition Language (DDL) and Data Manipulation Language (DML) statements. The query string and destination table can be parameterized, allowing you to organize query results by date and time.

Now you add a query that checks each day for new data. When new trees are planted, you'll get the additional stats updated directly into the `reports.trees` table.

Click **Compose New Query** and run the following query to pull incremental data into the reports.trees table on a daily basis using the scheduled query feature:

```
SELECT

 TIMESTAMP_TRUNC(plant_date, MONTH) as plant_month,
  COUNT(tree_id) AS total_trees,
  species,
  care_taker,
  address,
  site_info
FROM
  `bigquery-public-data.san_francisco_trees.street_trees`
WHERE
  address IS NOT NULL
  AND plant_date >= TIMESTAMP_SUB(CURRENT_TIMESTAMP(),
INTERVAL 1 DAY)
  AND plant_date < TIMESTAMP_TRUNC(CURRENT_TIMESTAMP(), DAY)
GROUP BY
  plant_month,
  species,
  care_taker,
  address,
  site_info
```

Click on the **Schedule** button, then **Create new scheduled query**:
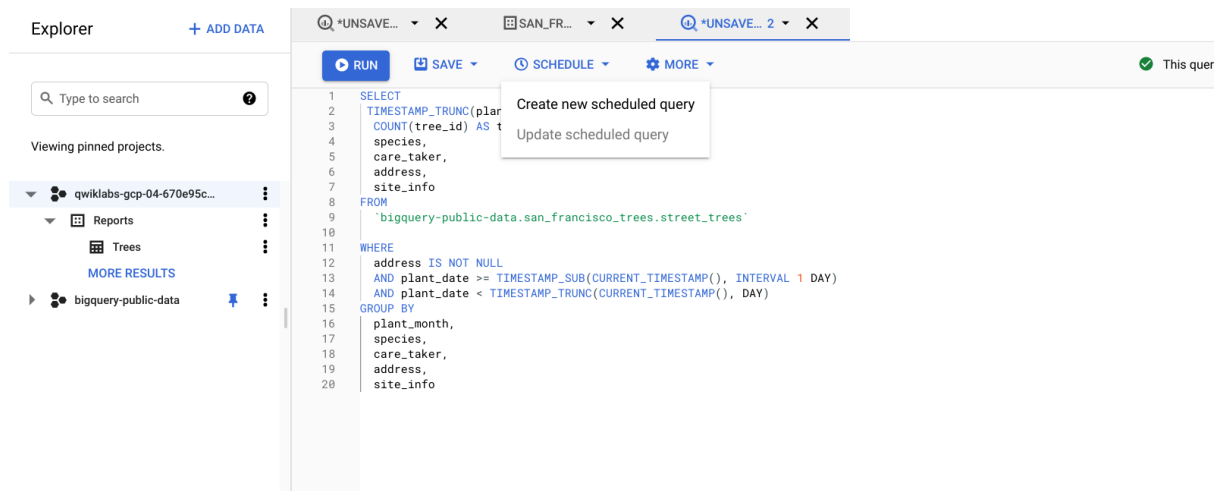
On the new Scheduled query page, set the following:

- Name: **Update_trees_daily**

Schedule options:

- Repeats: **daily**, choose date and time in the future

In the **Destination for query results** sections, your project name and dataset name are already selected for you.

- Table name: type in "Trees" and select **Append to table** so it doesn't overwrite existing data.

New scheduled query

Details and schedule

Name for scheduled query

Update_trees_daily

Schedule options

Repeats

Daily

○ Start now   ● Schedule start time

Start date and run time

12/22/19, 4:07 AM EST ▾

● End never   ○ Schedule end time

⚠ This schedule will run Every day at 04:07 America/New York, starting Sun Dec 22 2019

Destination for query results

ℹ A destination table is required to save scheduled query options.

Project name   Dataset name

qwiklabs-gcp-00-60fa33e0aea0 ▾   Reports ▾

Table name

Trees

Destination table write preference
● Append to table
○ Overwrite table

Advanced options ⌄

Notification options

☐ Send email notifications ❓

Cloud Pub/Sub topic ❓

projects/<project>/topics/<topic-name>

Schedule   Cancel

Click **Schedule**.

You may need to give your lab credentials permission, then agree to replace your query.

Note: If you run this query you won't see any new results because they haven't happened yet.

Create new data sources in Data Studio

Now you'll build your dashboard using the tree data you've just aggregated with Data Studio.

Open a new tab in your browser and go to Data Studio with this link:

https://datastudio.google.com.

Click **Create** in the top left, and then click **Report**.

Enter the country and check the terms and conditions. Click **Continue**

- Select **No, thanks** for all email offers, and then click Continue.

Create a new report in Data Studio

Click on the **BigQuery**, then click **Authorize**, then **Allow**.

Now you'll use the BigQuery connector to connect to the `reports.trees` table.

Start by selecting your **project**, then the **Reports** dataset, then the **Trees** table, as shown below:

click **Add** and then click **Add to Report**.

Now you can create charts using the data in this table.

Click **Add a Chart**,

Click on the **Add a chart dropdown** and select the type you want. In this example, you can see the following types of charts:

1. Stacked column bar graph showing the number of trees planted each month and the name of the caretaker who planted them.
2. A scorecard showing the total number of trees added in the last year.
3. A pie chart showing the percent distribution of trees planted by their species.
4. A table chart along with a bar graph representing the number of trees planted by site.
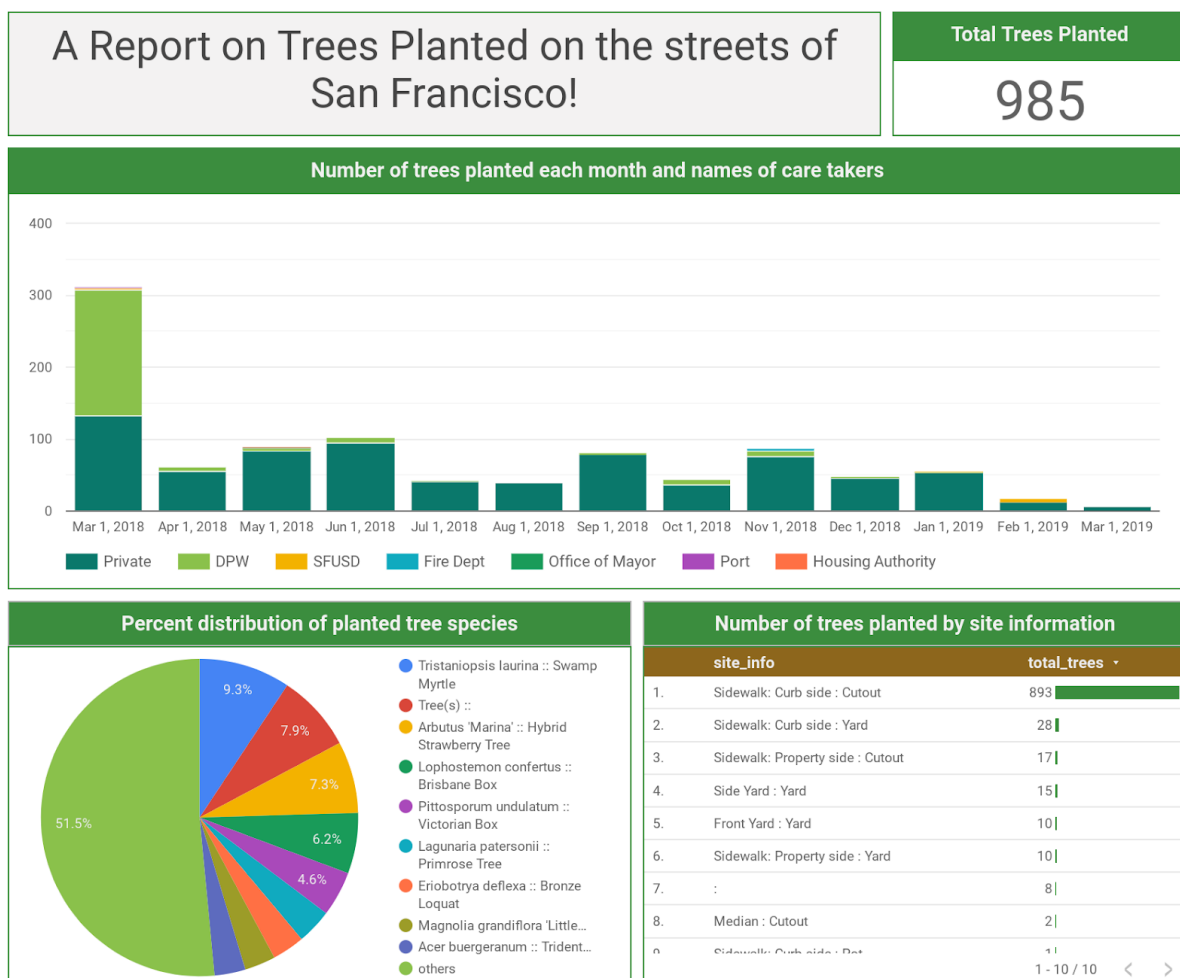
You can experiment on your own creating charts and titles modeled after the example. Here are some hints:

- Titles are created using the text tool. In the example, titles were created for each chart and the dashboard itself.
- When a chart is selected, you can edit the colors and font sizes when you click on the **Style** tab on the right-hand side.
- Click on a chart to modify its size and drag it to a new location.