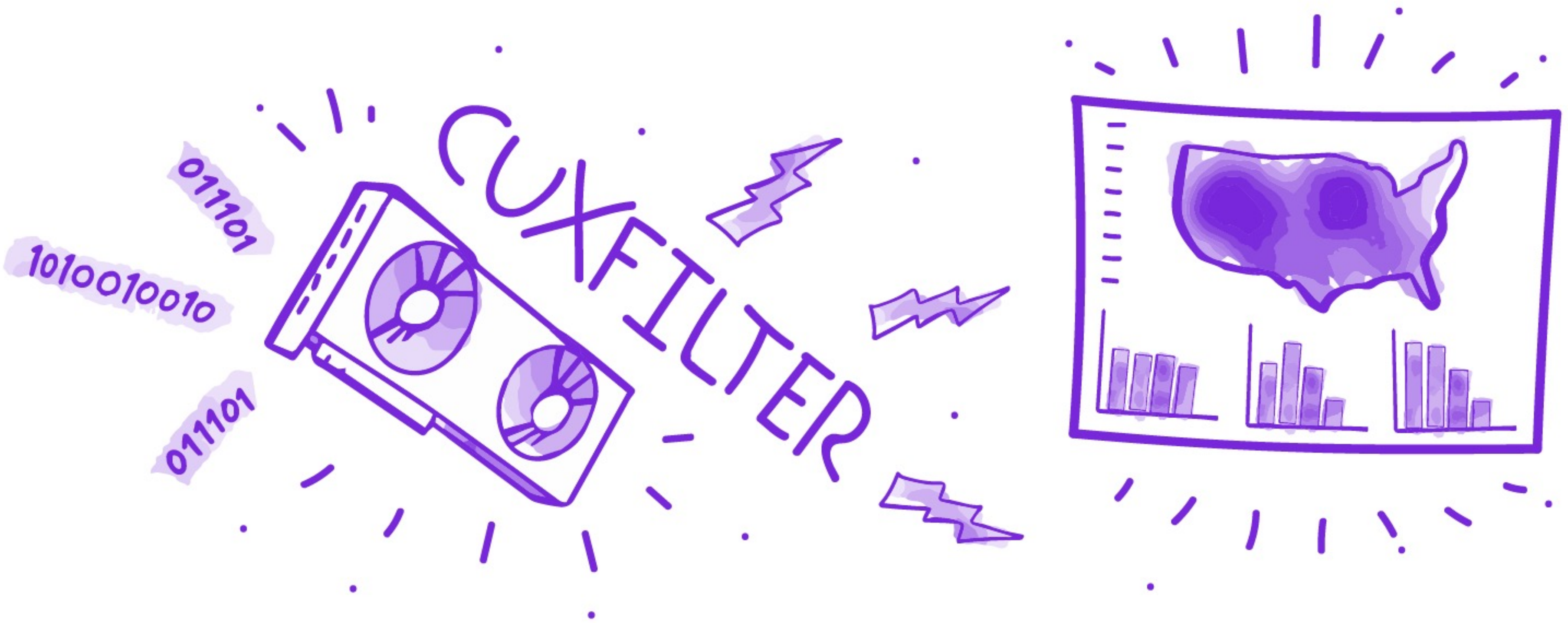
DLI Accelerated Data Science Teaching Kit

Lecture 7.6 - cuXFilter

How to implement visualization?



cuXFilter

(ku-cross-filter)

Retrieved from https://github.com/rapidsai/cuxfilter

What is cuXFilter?

- RAPIDS framework connecting web visualization to GPU accelerated cross-filtering (filtering data and having visualization tool reflect the changes)

- Inspired from Crossfilter library

- What is Crossfilter?
  - Javascript library
  - Explore multivariate datasets in browser with filtering mechanism

- Limitations of Crossfilter?
  - Data stored in client-side browser memory
  - Inefficient on large datasets

How to overcome limitations associated with large datasets?

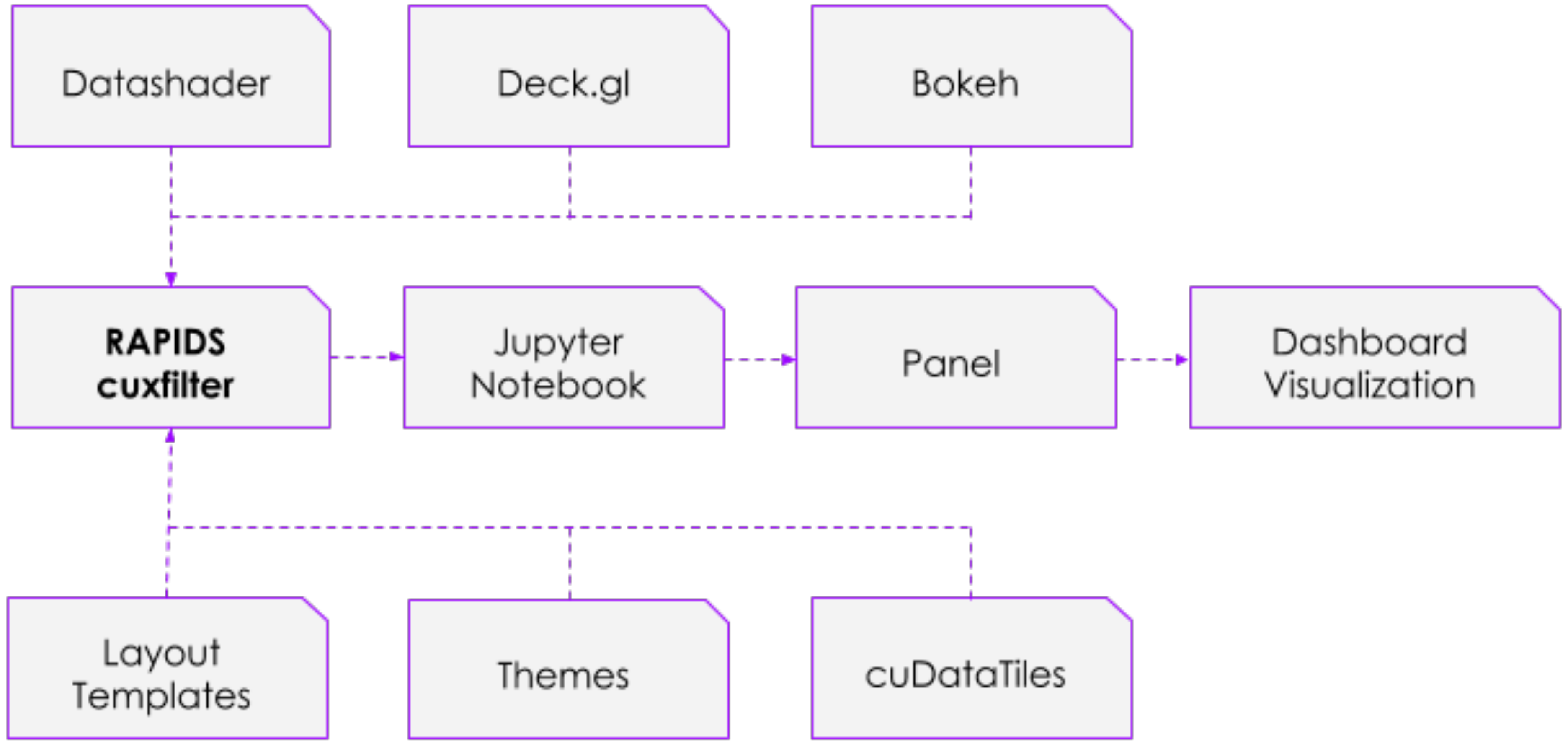
Maintain data in GPU via RAPIDS’ GPU Dataframe (cuDF)

No longer dependent on browser memory

RAPIDS GPU acceleration now available

What Does cuXFilter Do?

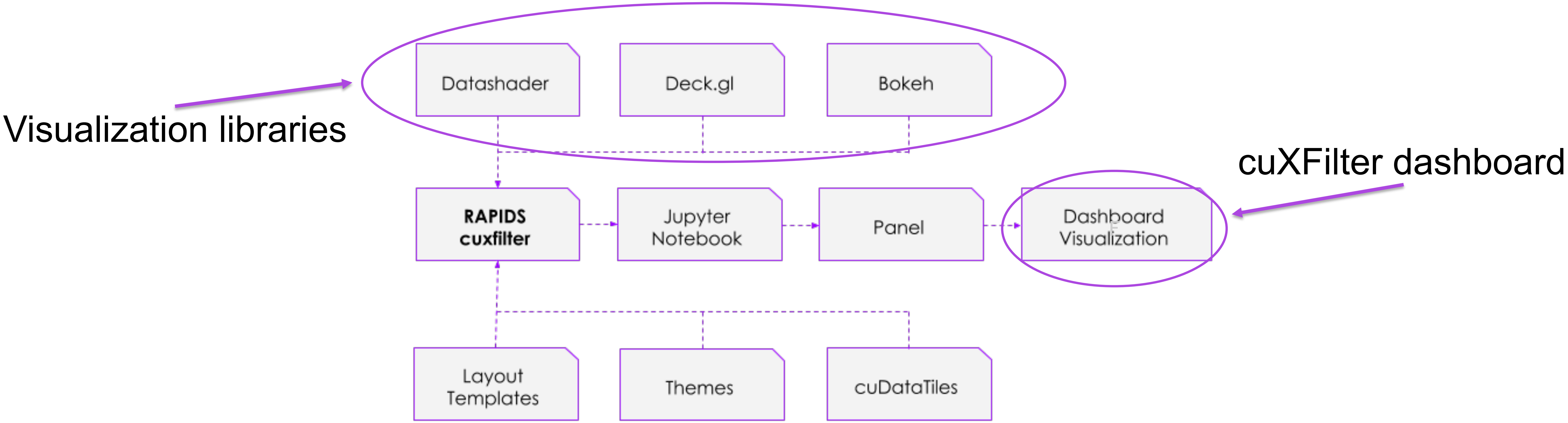
- Connects different visualization libraries to GPU backend
- Provides user with UI components connecting libraries
- Precomputes aggregations for quick visualization interactions via cuDataTiles



Retrieved from https://github.com/rapidsai/cuxfilter

What Does cuXFilter Do?

- Connects different visualization libraries to GPU backend
- Provides user with UI components connecting libraries
- Precomputes aggregations for quick visualization interactions via cuDataTiles



Retrieved from https://github.com/rapidsai/cuxfilter

How fast is cuXFilter?

cuDataTiles take around 250 milliseconds per chart given a 100 million row dataset.

How fast is cuXFilter?

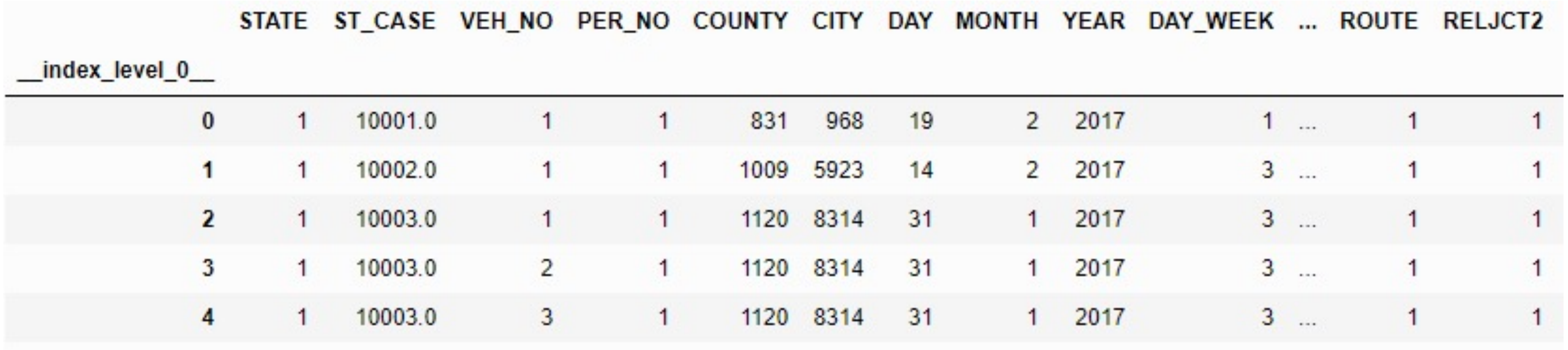
cuDataTiles take around 250 milliseconds per chart given a 100 million row dataset.

Quite fast!

cuXFilter Demonstration

Importing Data and Setting Up cuDF

```
import cuxfilter

cux_df = cuxfilter.DataFrame.from_dataframe("./path_to_dataframe")

#would display something like image from below
cux_df.data.head()
```



	STATE	ST_CASE	VEH_NO	PER_NO	COUNTY	CITY	DAY	MONTH	YEAR	DAY_WEEK	...	ROUTE	RELJCT2
__index_level_0__													
0	1	10001.0	1	1	831	968	19	2	2017	1	...	1	1
1	1	10002.0	1	1	1009	5923	14	2	2017	3	...	1	1
2	1	10003.0	1	1	1120	8314	31	1	2017	3	...	1	1
3	1	10003.0	2	1	1120	8314	31	1	2017	3	...	1	1
4	1	10003.0	3	1	1120	8314	31	1	2017	3	...	1	1

Retrieved from https://docs.rapids.ai/api/cuxfilter/stable/10%20minutes%20to%20cuxfilter.html

Creating Charts Based Off Data

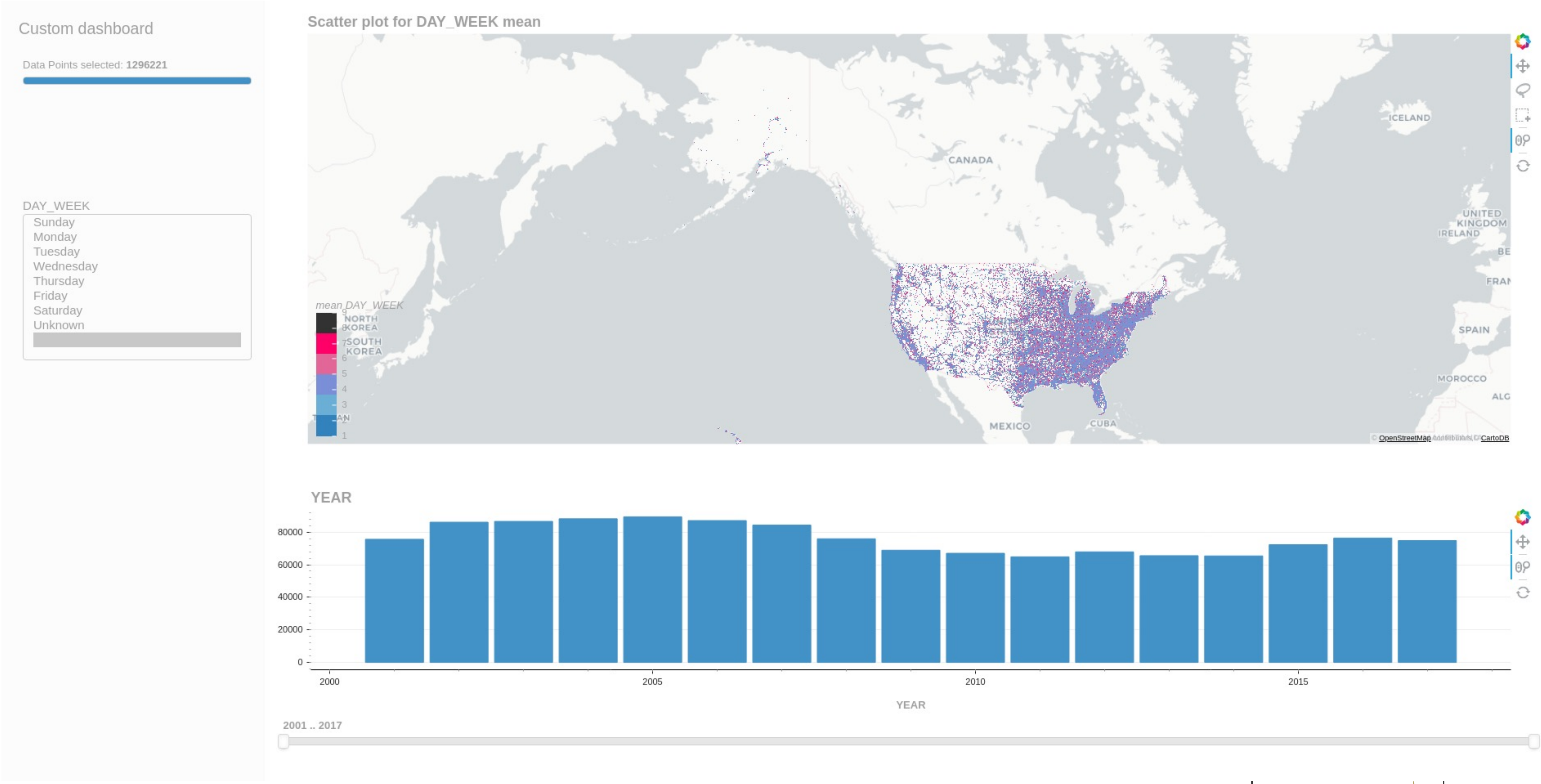
```
demo_palette = ["#3182bd", "#6baed6", "#7b8ed8", ... , "#323232" ]

chart1 = cuxfilter.charts.scatter(x='data_column_1',
   y='data_column_2', aggregate_col='data_column_3',
   aggregate_fn='mean', color_palette=demo_palette,
   tile_provider='CARTODBPOSITRON’, pixel_shade_type='linear’)

chart2 = cuxfilter.charts.bar('data_column_4')


...

charts_list = [chart1, chart2, ...]
```

Retrieved from https://docs.rapids.ai/api/cuxfilter/stable/10%20minutes%20to%20cuxfilter.html

Viewing Dashboard with Charts

```
d = cux_df.dashboard(charts_list, title='Custom dashboard’,
    layout=layouts.feature_and_base, theme=themes.light, data_size_widget=True)
```

```
wait d.preview()
```



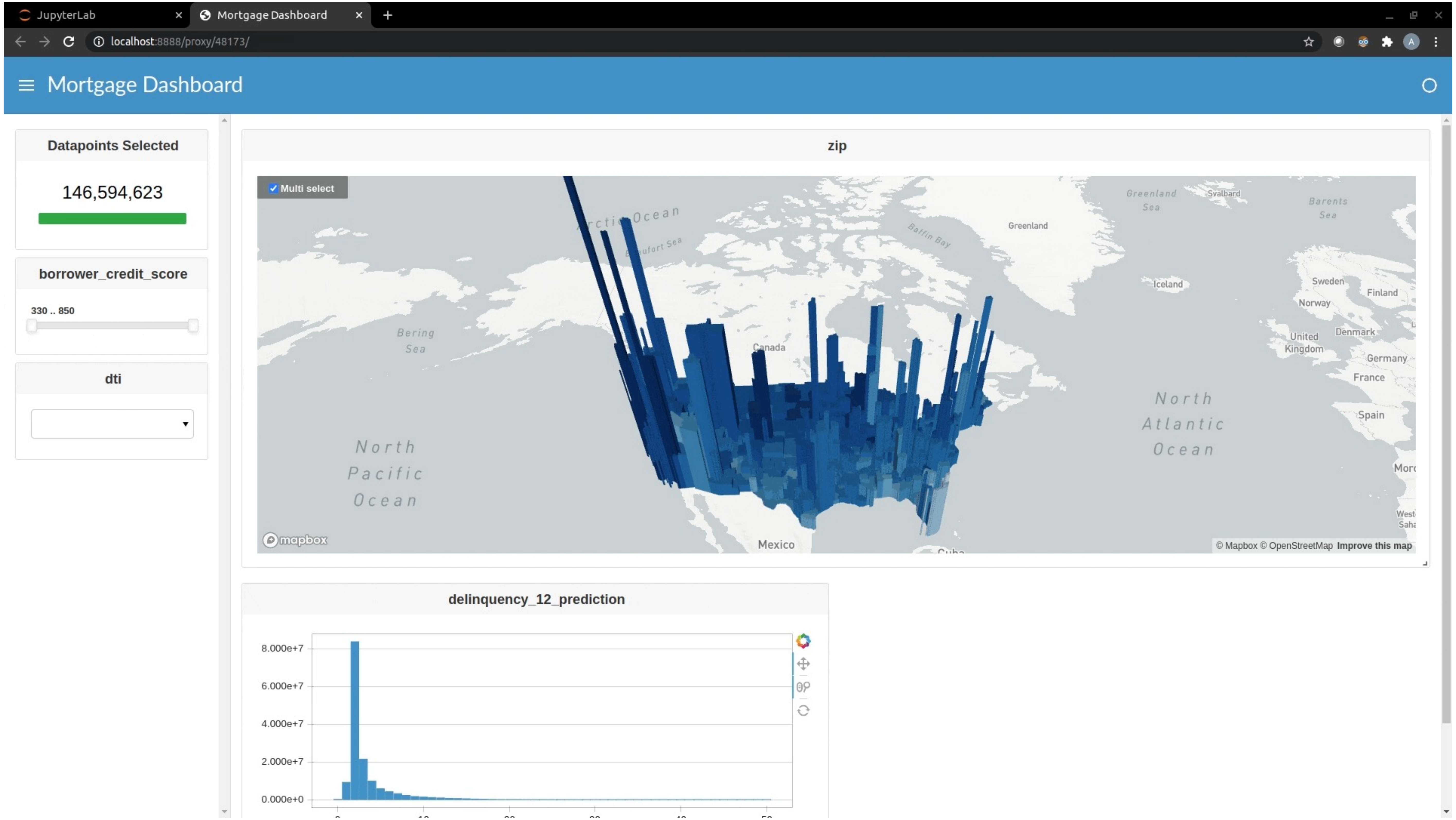
Retrieved from https://docs.rapids.ai/api/cuxfilter/stable/10%20minutes%20to%20cuxfilter.html

Running Dashboard and Saving Snapshot

```
#opens in new tab
d.show('current_notebook_url:current_notebook_port')

#saves snapshot as dataframe
current_state_df = d.export()

#stops server
d.stop()
```

Retrieved from https://docs.rapids.ai/api/cuxfilter/stable/10%20minutes%20to%20cuxfilter.html

Live Demo of cuXFilter Dashboard



JupyterLab | Mortgage Dashboard

localhost:8888/proxy/48173/

Mortgage Dashboard

Datapoints Selected

146,594,623

borrower_credit_score

330 .. 850

dti

zip

☑ Multi select

delinquency_12_prediction

Demo retrieved from https://github.com/rapidsai/cuxfilter

DEEP LEARNING INSTITUTE

Georgia Tech

PRAIRIE VIEW A&M UNIVERSITY

DLI Accelerated Data Science Teaching Kit

Thank You