Letter to Petersen about graphs:

Hello, Professor Petersen!

I have discovered that there are two different ways I can graph a chart with a web application.

I can either use a client-side JavaScript library to do graphical processing, or I can use the server-side matplotlib library from Python to do graphical processing.

JavaScript(client-side):

Advantages:

* Data is available locally for reprocessing
* Once the .js and .css libraries are downloaded they can be re-used with no further download overhead

Disadvantages:

* Multiple identical images need to be separately drawn
* .js and .css libraries need to be part of the download package
* JavaScript must be available and activated on the client
* Taking a copy of a chart requires a screenshot
* Libraries load without screen activity, giving the appearance of slow loading

Matplotlib(server-side):

Advantages:

* Data remains confidential
* Chart can be copied and pasted as a separate unit
* If multiple copies of the chart are required, the image can be easily repeated
* Browser may be able to resize the image
* Images load progressively, visibly in the browser

Disadvantages:

* Identical data cannot be reprocessed without refreshing the page
* Heavier load on the server processor
* Server-side image storage required

I have been using the matplotlib library so far to do my graphing, but I fear that if more than 5 people use our server simultaneously, the server might crash.

I recommend that we use a client-side graphing system to reduce the load on our server, so each client processes their own chart, and our server just serves the data required to make the chart.

Sources: <https://www.linux.com/news/server-or-client-processing-better-charts-and-graphs/>