

# Twitter Filtering System

# NETFLIX

Rupesh Dabbir

---



- A Project report By Rupesh Dabbir

(And with a cup of Coffee over weekend ;-)


---

# WALKTHROUGH:

## **Build System / Technology Stack:**

- Written in React / Node.js / Webpack / Babel
- A customizable Filter Engine is written in React.js.
- Client-side rendered.
- CI / CD Over Heroku
- Paginated and configurable Tweet Dashboard.

# CHALLENGES:

- There are numerous challenges I've faced while solving this problem. However, these problems really made me introspect and think about various design choices!
  - Since we are listening to server-sent events, handling of data is the biggest challenge to avoid UI freeze (Since JS is single-threaded).
  - I have solved this problem using Async data fetching and using Priority Queue for making sure I show (and keep) date information.
  - Filtering engine was another challenge, designing a parser that could parse all the rules was fun!
  - Making a filtering engine extensible for "N" parameters for future proof was another challenge.
- 

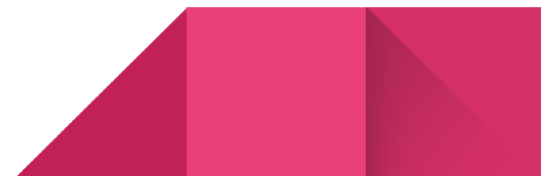
---

# ARCHITECTURE:

There are a few key challenges that I have encountered while solving this problem.

1. UI was continuously freezing due to the fact Server-Sent Events are always continuous and opening up Event Source always would be inviting devil home :-)
2. This problem has been solved in a few layers. The following diagram will give more insight(s) on how I decided to tackle the problem:

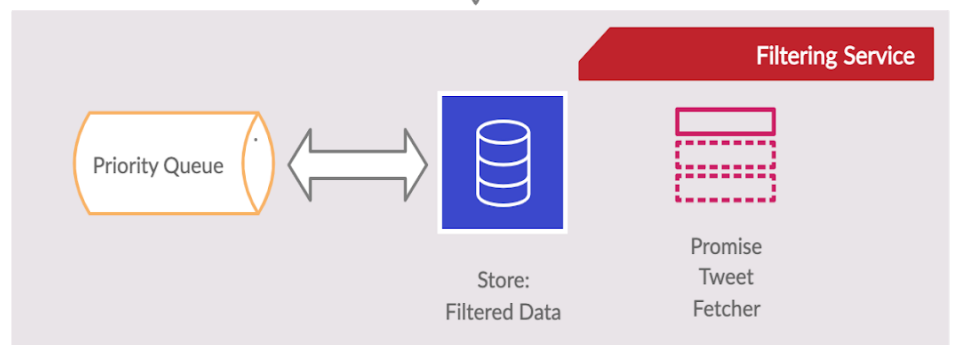
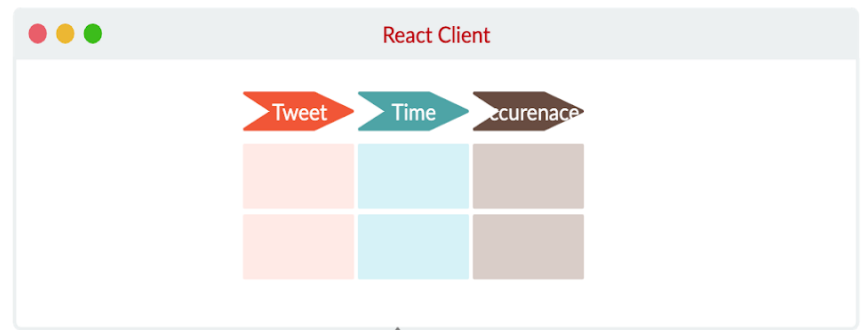
Please find the architectural diagram below:





Twitter Tweet Filter Service

Twitter Server-Sent Events



There are several key components in this architecture: (Component-based architecture)

## I. Dashboard Component:

| Twitter Live Filtering Service           |         |               |               |          |          |  | GitHub | Add Filter |
|--|---------|---------------|---------------|----------|----------|--|--------|------------|
| Tweet                                    | User    | Retweet Count | Created At    | Verified | Language |  |        |            |
| daredevil cool. #newepisode              | user-63 | 478           | 1568611442060 |          | es       |  |        |            |
| cuervos cool. #netflexing                | user-48 | 88            | 1568611442061 |          | fr       |  |        |            |
| busted data: (" cool. #greatshow         | user-91 | 110           | 1568611442061 |          | en       |  |        |            |
| kublai khan cool. #greatshow             | user-38 | 520           | 1568611442061 |          | de       |  |        |            |
| ointb meh. #newepisode                   | user-19 | 379           | 1568611442061 |          | de       |  |        |            |
| kublai khan boring. #bingewatching       | user-1  | 282           | 1568611442061 |          | en       |  |        |            |
| daredevil awesome. #trailers             | user-81 | 563           | 1568611442061 |          | en       |  |        |            |
| stranger things boring. #newseason       | user-18 | 665           | 1568611442061 |          | fr       |  |        |            |
| house of cards cool. #binging            | user-10 | 778           | 1568611442061 |          | en       |  |        |            |
| narcos rocks. #newepisode                | user-65 | 153           | 1568611442061 |          | de       |  |        |            |
| daredevil meh. #trailers                 | user-62 | 673           | 1568611442061 |          | fr       |  |        |            |
| daredevil sucks. #greatshow              | user-72 | 843           | 1568611442061 |          | es       |  |        |            |
| orange is the new black rocks. #newse... | user-82 | 48            | 1568611442061 |          | de       |  |        |            |
| kublai khan awesome. #greatshow          | user-49 | 383           | 1568611442061 |          | fr       |  |        |            |
| marco polo rocks. #trailers              | user-34 | 362           | 1568611442061 |          | es       |  |        |            |
| house of cards cool. #bingewatching      | user-90 | 841           | 1568611442061 |          | de       |  |        |            |
| marco polo cool. #newrelease             | user-94 | 96            | 1568611442061 |          | fr       |  |        |            |
| narcos amazing. #netflexing              | user-41 | 42            | 1568611442061 |          | en       |  |        |            |
| cuervos sucks. #netflexing               | user-83 | 343           | 1568611442061 |          | es       |  |        |            |
| house of cards sucks. #netflixandchill   | user-73 | 472           | 1568611442061 |          | de       |  |        |            |
| Previous                                 |         | Page 1 of 350 | 20 rows       | Next     |          |  |        |            |

- Responsible for displaying the high volume of tweets that are coming in while keeping UI stable. This is done through the following Optimization techniques:
  1. I used a **Priority Queue** (Max-Heap) to keep track of the latest tweets.
    - a. I wrote a custom comparator that keeps track of "latest" tweets in comparison to the Time-stamp of the tweet.
  2. An **Async** Promise runner that keeps track of the incoming tweets (To make sure it doesn't crash the browser) and we

---

keep incrementing the tweets by 100 for every Configurable time (For now it's 10 seconds)

3. **Retry component** that retries asynchronously without blocking the page, fetches N+100 tweets and then closes the connection (to avoid unnecessary overhead on the browser and freezing it)

## II. Filter Engine:

The screenshot shows a modal dialog titled "Please make your filter criteria and click Save!". Inside the dialog, there is a filter rule editor. It starts with "And" and a "+" sign. Below this, there are two filter criteria: "tweet" followed by "Contains" and a text input field containing "narcos". The second criterion is "tweet" followed by "Equals" and an empty text input field. A dropdown menu is open under the second "tweet" label, showing options "tweet" and "lang". At the bottom left of the dialog is a "Save" button. The dialog has a close button (X) in the top right corner.

- A dynamic filter component that accepts various parameters such as "Contains", "Equals" and RegEx and returns appropriate match.
- Filter Algorithm:
  - Takes in bunch of arguments and parameters from user.

- Compiles a list of "actionable" filter keys that needs to be validated across the data in Priority Queue based on user entered input (RegEx, Contains, Equals etc)
- The Filter component is **EXTENDABLE** for any number of parameters and any inputs.

### III. Table Component:

| Twitter Live Filtering Service           |         |               |               |          |          | GitHub | Add Filter |
|--|---------|---------------|---------------|----------|----------|--------|------------|
| Tweet                                    | User    | Retweet Count | Created At    | Verified | Language |        |            |
| daredevil cool. #newepisode              | user-63 | 478           | 1568611442060 |          | es       |        |            |
| cuervos cool. #netflexing                | user-48 | 88            | 1568611442061 |          | fr       |        |            |
| busted data: (" cool. #greatshow         | user-91 | 110           | 1568611442061 |          | en       |        |            |
| kublai khan cool. #greatshow             | user-38 | 520           | 1568611442061 |          | de       |        |            |
| ointb meh. #newepisode                   | user-19 | 379           | 1568611442061 |          | de       |        |            |
| kublai khan boring. #bingewatching       | user-1  | 282           | 1568611442061 |          | en       |        |            |
| daredevil awesome. #trailers             | user-81 | 563           | 1568611442061 |          | en       |        |            |
| stranger things boring. #newseason       | user-18 | 665           | 1568611442061 |          | fr       |        |            |
| house of cards cool. #binging            | user-10 | 778           | 1568611442061 |          | en       |        |            |
| narcos rocks. #newepisode                | user-65 | 153           | 1568611442061 |          | de       |        |            |
| daredevil meh. #trailers                 | user-62 | 673           | 1568611442061 |          | fr       |        |            |
| daredevil sucks. #greatshow              | user-72 | 843           | 1568611442061 |          | es       |        |            |
| orange is the new black rocks. #newse... | user-82 | 48            | 1568611442061 |          | de       |        |            |
| kublai khan awesome. #greatshow          | user-49 | 383           | 1568611442061 |          | fr       |        |            |
| marco polo rocks. #trailers              | user-34 | 362           | 1568611442061 |          | es       |        |            |
| house of cards cool. #bingewatching      | user-90 | 841           | 1568611442061 |          | de       |        |            |
| marco polo cool. #newrelease             | user-94 | 96            | 1568611442061 |          | fr       |        |            |
| narcos amazing. #netflexing              | user-41 | 42            | 1568611442061 |          | en       |        |            |
| cuervos sucks. #netflexing               | user-83 | 343           | 1568611442061 |          | es       |        |            |
| house of cards sucks. #netflixandchill   | user-73 | 472           | 1568611442061 |          | de       |        |            |
| Previous                                 |         | Page 1 of 350 | 20 rows       | Next     |          |        |            |

- Showing data in perfect tabular format for better viewing experience.
- Paginated for easier viewing of data.
- Can display #of records per page.

### IV. Modal Component:

- Responsible for rendering Modal that shows Filter component.

# Continuous Integration:

The screenshot shows the Heroku dashboard for a project named 'twitter-data-fetcher'. The breadcrumb navigation indicates the path: Personal > twitter-data-fetcher. Below this, it shows the GitHub repository 'rupeshdabbir/twitter-server-sent' and the 'master' branch. The main navigation bar includes links for Overview, Resources, Deploy, Metrics, Activity (which is the active tab), Access, and Settings. The 'Activity Feed' section displays a list of recent events:

- Deployed** (9bf31943) by d.rupeshkumar@gmail.com at 12:30 PM · v3.
- Build succeeded** by d.rupeshkumar@gmail.com at 12:28 PM. A link to [View build log](#) is provided.
- Enable Logplex** by d.rupeshkumar@gmail.com at 12:26 PM · v2. A link to [Roll back to here](#) is provided.
- Initial release** by d.rupeshkumar@gmail.com at 12:26 PM · v1. A link to [Roll back to here](#) is provided.

- Integrated the project using Heroku setup!
- It's a one tap process after push, generates the build artifact and verifies if everything looks good and pushes to production.




---

## RUNNING THE APP:

- Setup / Running is pretty simple:
  - Pull the code from <https://github.com/rupeshdabbir/twitter-server-sent> or from zip.
  - npm install
  - npm run start for starting the server.
  - Demo Running Live: <https://twitter-data-fetcher.herokuapp.com/>

If you encounter any issues, please delete your node\_modules and try again. No need for any database connections or setup :-)

## IMPROVEMENTS:

- I could do a lot of improvements if I had time. To start off, implement a "Clear Filter" feature that would clear the filter.
  - Currently, there is no EDIT filter option, so the user would need to refresh the page and try to add the filter again. (Due to the interest of the time)
- 

- 
- I could do the processing on server side and establish connection between client and server using GRPC.
  - Can use Splunk to chunk the data on server-side. Keep a big priority queue and keep going!