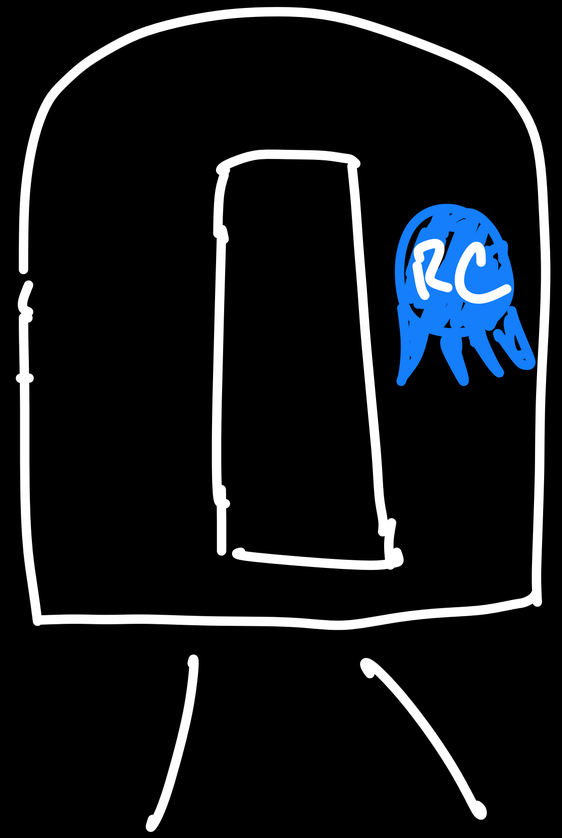
subway surfing

or, my initial adventures with the MTA API



P.B. To
The Recurse Center
Jan 25 2024

“countdown clocks”

https://medium.com/good-service/new-feature-detecting-delays-48d29df9ba54

MTA

Docs Feeds Access Key Profile Sign Out

/ Realtime Data Feeds

Realtime Data Feeds

MTA open data – including train current service status and elevator and escalator status are provided free of charge to developers who agree to the MTA's Terms and Conditions of Use.

Subway Feeds

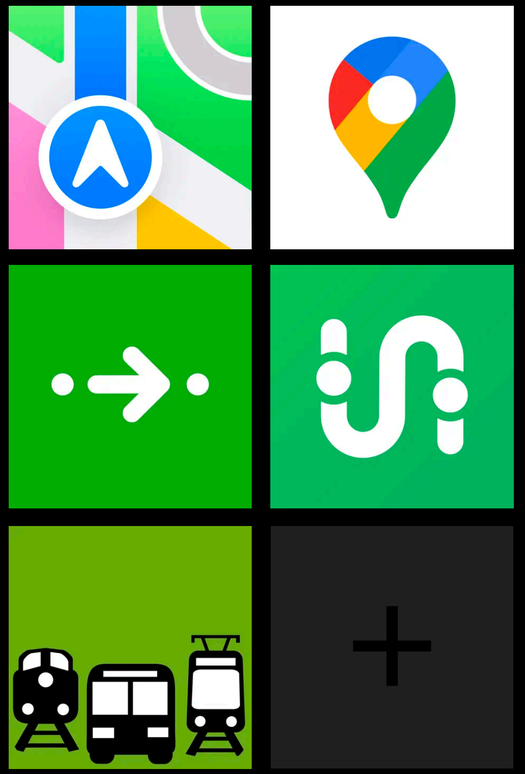
LIRR feeds

MNR feeds

Service Alerts

Elevator and Escalator

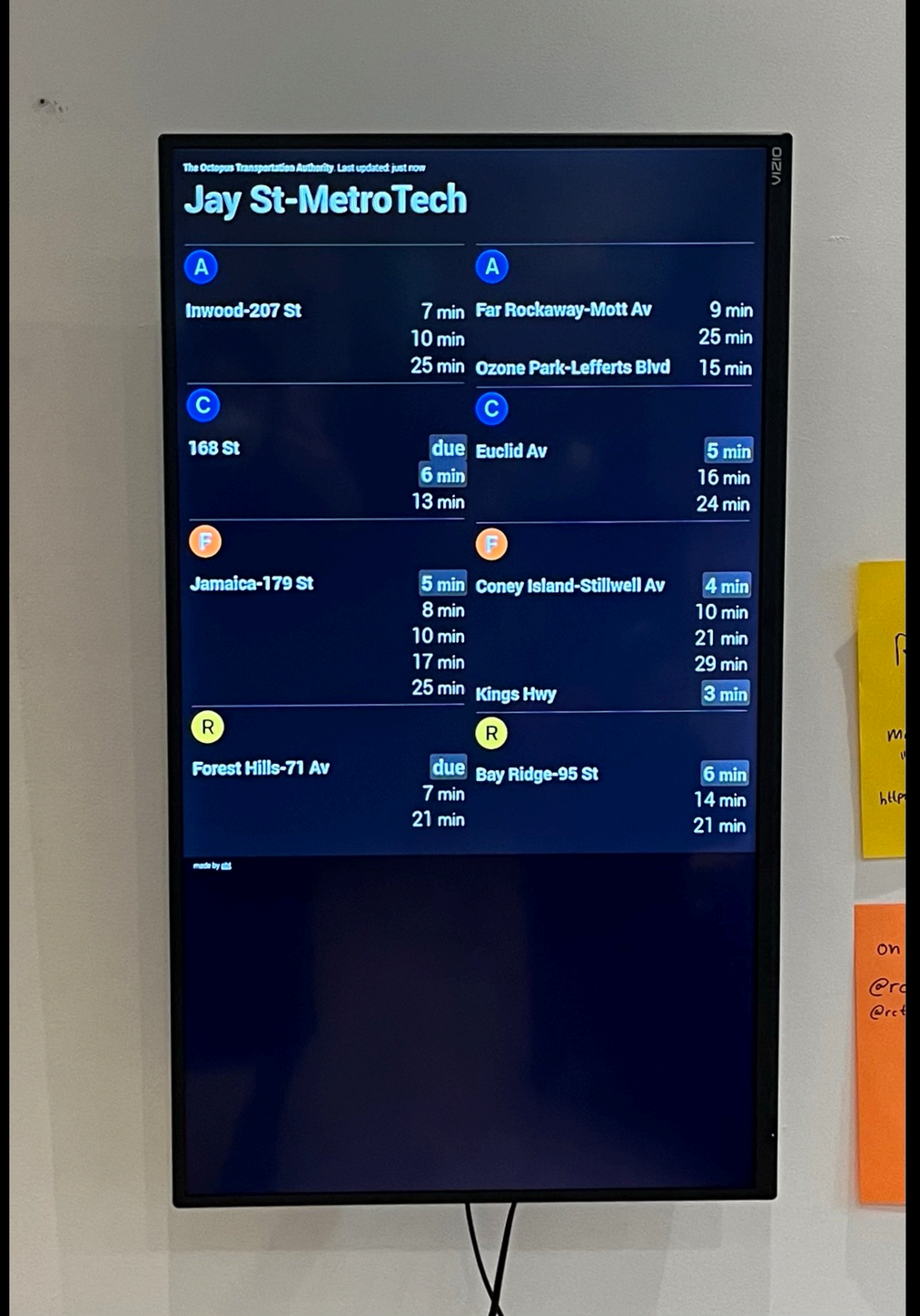
https://api.mta.info/#/landing



so i made this

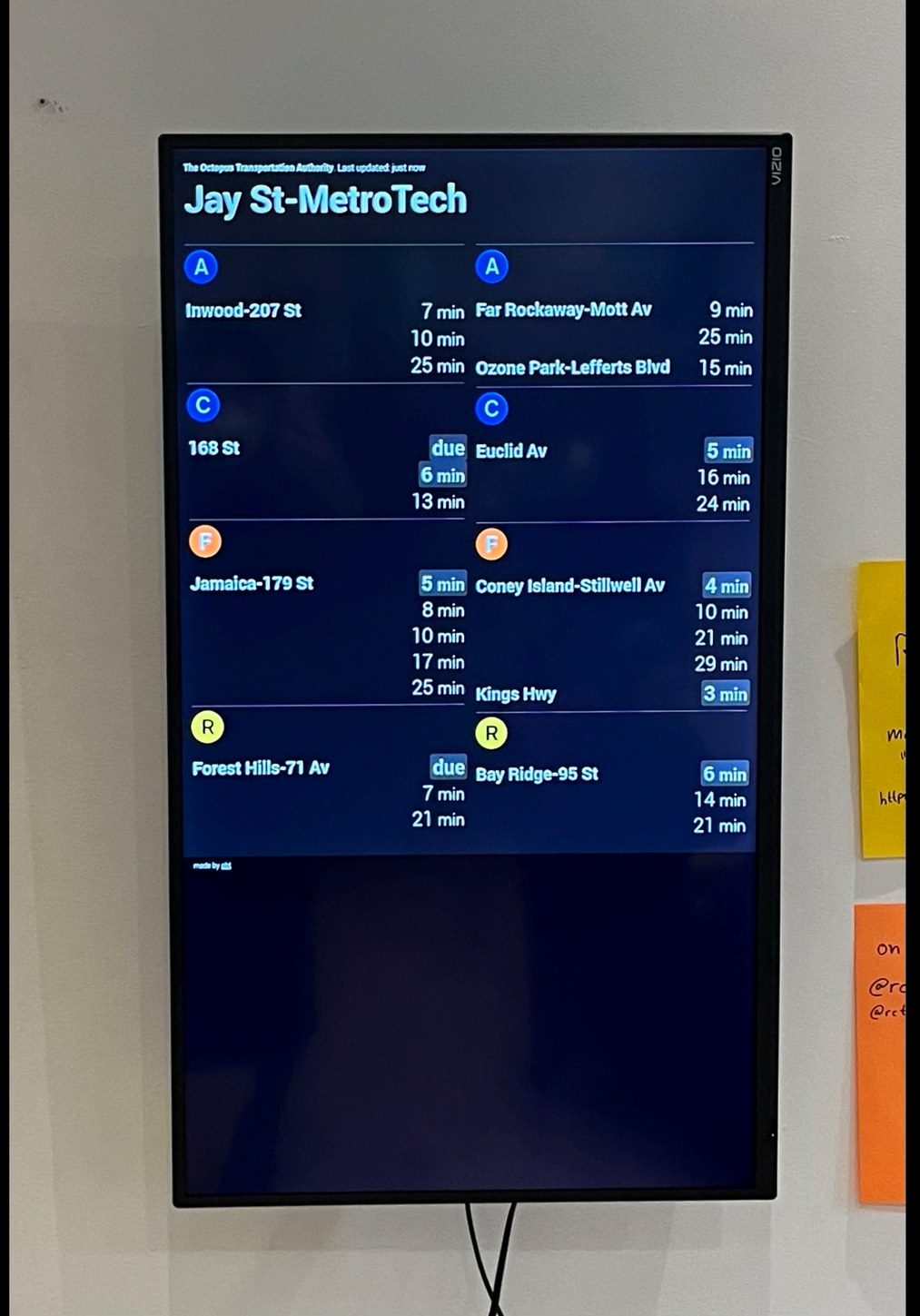
it's a departure board/ countdown clock for the MTA running on RCTV

https://ota.pbt.dev/



Goals

1. Learn about the MTA API
2. Make it readable
3. Make it run fast



What I learned about the MTA's API

1. This protobuf file has most of the documentation

```
// Realtime update for arrival and/or departure events for a given stop on a
// trip. Updates can be supplied for both past and future events.
// The producer is allowed, although not required, to drop past events.
message StopTimeUpdate {
  // The update is linked to a specific stop either through stop_sequence or
  // stop_id, so one of the fields below must necessarily be set.
  // See the documentation in TripDescriptor for more information.

  // Must be the same as in stop_times.txt in the corresponding GTFS feed.
  optional uint32 stop_sequence = 1;
  // Must be the same as in stops.txt in the corresponding GTFS feed.
  optional string stop_id = 4;

  optional StopTimeEvent arrival = 2;
  optional StopTimeEvent departure = 3;

  // The relation between this StopTime and the static schedule.
  enum ScheduleRelationship {
    // The vehicle is proceeding in accordance with its static schedule of
    // stops, although not necessarily according to the times of the schedule.
    // At least one of arrival and departure must be provided. If the schedule
    // for this stop contains both arrival and departure times then so must
    // this update.
    SCHEDULED = 0;

    // The stop is skipped, i.e., the vehicle will not stop at this stop.
    // Arrival and departure are optional.
    SKIPPED = 1;

    // No data is given for this stop. The main intention for this value is to
    // give the predictions only for part of a trip, i.e., if the last update
    // for a trip has a NO_DATA specifier, then StopTimes for the rest of the
    // stops in the trip are considered to be unspecified as well.
    // Neither arrival nor departure should be supplied.
    NO_DATA = 2;
  }
  optional ScheduleRelationship schedule_relationship = 5
      [default = SCHEDULED];

  // The extensions namespace allows 3rd-party developers to extend the
  // GTFS Realtime Specification in order to add and evaluate new features
  // and modifications to the spec.
  extensions 1000 to 1999;
}
```

Thankfully, Python bindings exist

Trains arriving at Jay Street - MetroTech

```
[2]  from IPython.display import display
     from nyct_gtfs import NYCTFeed
     from nyct_gtfs.gtfs_static_types import Stations
     from itertools import groupby

     from datetime import datetime, date, time, timezone, tzinfo, timedelta

     ApiKey = [redacted]
     nyctime = timezone(-timedelta(hours=5))

     feedACE = NYCTFeed("A", api_key=ApiKey)
     feedF = NYCTFeed("F", api_key=ApiKey)
     feedR = NYCTFeed("R", api_key=ApiKey)
     stations = Stations()

     trains = []
     stops = ["A41N", "A41S", "R29N", "R29S"]

     for stop in stops:
       for feed in [feedACE, feedF, feedR]:
         trains += feed.filter_trips(headed_for_stop_id=stop, underway=True)

     arrivals = [[train.route_id, train.headsign_text if train.headsign_text else train.shape_id, [update.arrival for update in train.stop_time_updates if any
     relative_arrivals = [[route, dest, arrival_time, (arrival_time - datetime.now()).seconds // 60] for [route, dest, arrival_time] in arrivals]

     display(relative_arrivals)
```

```
['A', 'Far Rockaway-Mott Av', datetime.datetime(2024, 1, 25, 20, 6, 42), 10],
['A',
 'Ozone Park-Lefferts Blvd',
 datetime.datetime(2024, 1, 25, 20, 17, 20),
 21],
['A',
 'Ozone Park-Lefferts Blvd',
 datetime.datetime(2024, 1, 25, 20, 32, 20),
 36],
['A', 'Far Rockaway-Mott Av', datetime.datetime(2024, 1, 25, 20, 21, 43), 25],
['A', 'Far Rockaway-Mott Av', datetime.datetime(2024, 1, 25, 20, 36, 3), 40],
['C', 'Euclid Av', datetime.datetime(2024, 1, 25, 19, 59, 52), 3],
['C', 'Euclid Av', datetime.datetime(2024, 1, 25, 20, 8, 50), 12],
['C', 'Euclid Av', datetime.datetime(2024, 1, 25, 20, 21, 50), 25],
['C', 'Euclid Av', datetime.datetime(2024, 1, 25, 20, 27, 20), 31],
['C', 'Euclid Av', datetime.datetime(2024, 1, 25, 20, 40, 30), 44],
['F',
 'Coney Island-Stillwell Av',
```

https://github.com/Andrew-Dickinson/nyct-gtfs

https://flask.palletsprojects.com/en/3.0.x/

What I learned about the MTA's API

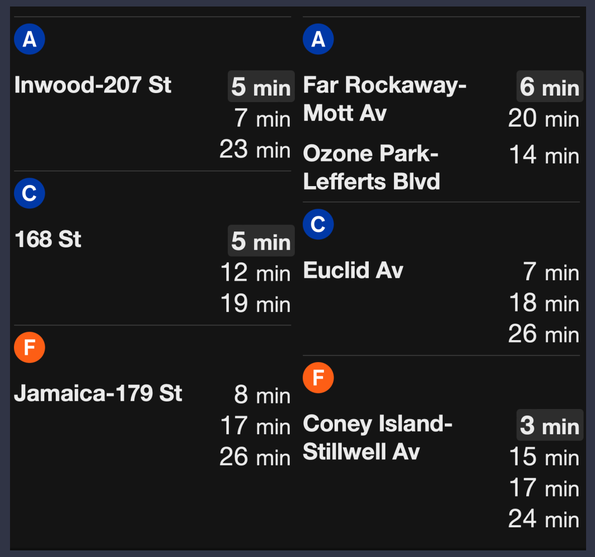
2. There are a lot of feeds

```
@cachetools.func.ttl_cache(maxsize=2, ttl=15)
def get_arrivals(stations):
    print(str(arrow.utcnow()), "getting arrivals")

    feeds = (
        NYCTFeed("1", api_key=ApiKey),
        NYCTFeed("A", api_key=ApiKey),
        NYCTFeed("F", api_key=ApiKey),
        NYCTFeed("G", api_key=ApiKey),
        NYCTFeed("J", api_key=ApiKey),
        NYCTFeed("L", api_key=ApiKey),
        NYCTFeed("N", api_key=ApiKey),
    )
```

What I learned about the MTA's API

3. A "stop" is just part of a station



A41

+

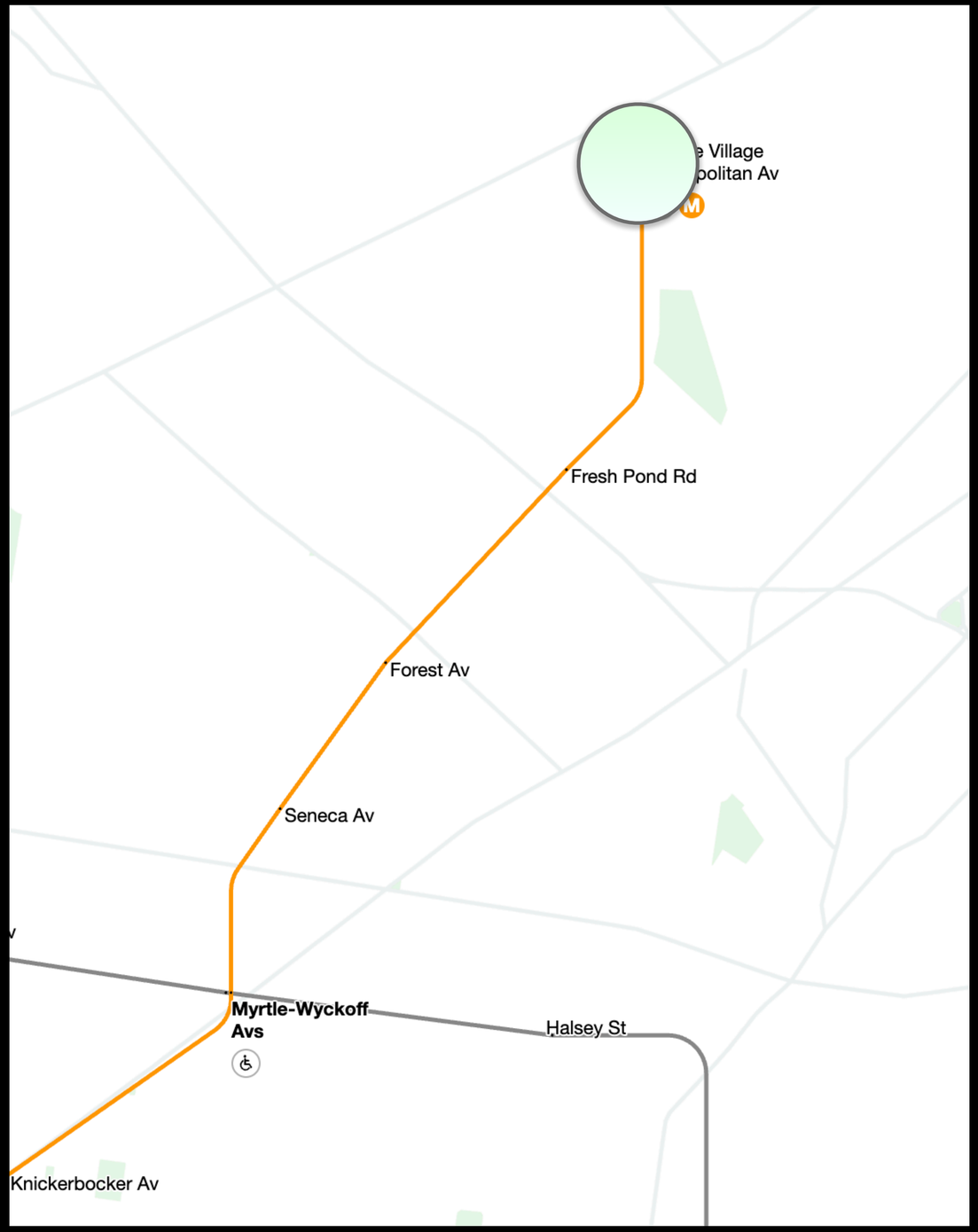
R29

=

What I learned about the MTA's API

4. If the train hasn't actually left, it doesn't exist*

*in this feed anyway


Fresh Pond Rd
Forest Av
Seneca Av
Myrtle-Wyckoff Avs
Halsey St
Knickerbocker Av

What I learned about the MTA's API

5. There are exactly two (2) directions

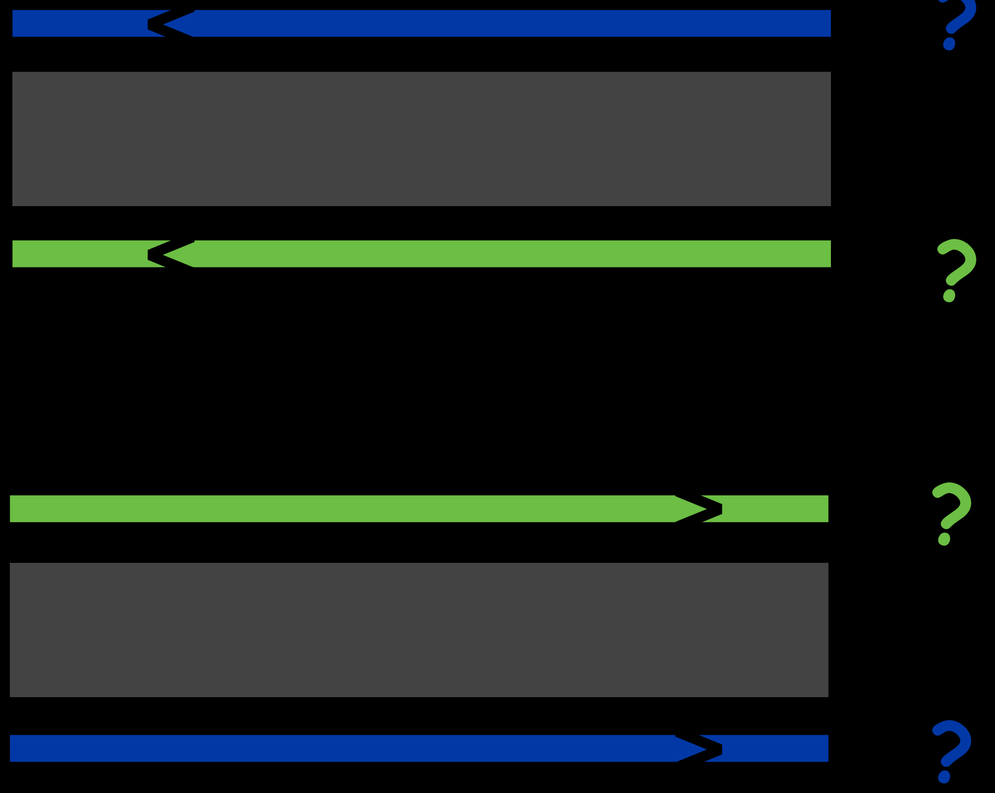
Hoyt-Schermerhorn St

A C **Manhattan**

G **Church Av** (Brooklyn)

G **Court Sq** (Queens)

A C **Queens**



Hoyt-Schermerhorn St

A C **Manhattan**

G **Church Av** (Brooklyn)

G **Court Sq** (Queens)

A C **Queens**

making it run fast*

* if it's warmed up...

https://ota.pbt.dev/

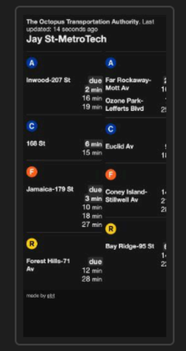
Performance	Accessibility	Best Practices	SEO	PWA
100	96	95	92	PWA

100

Performance

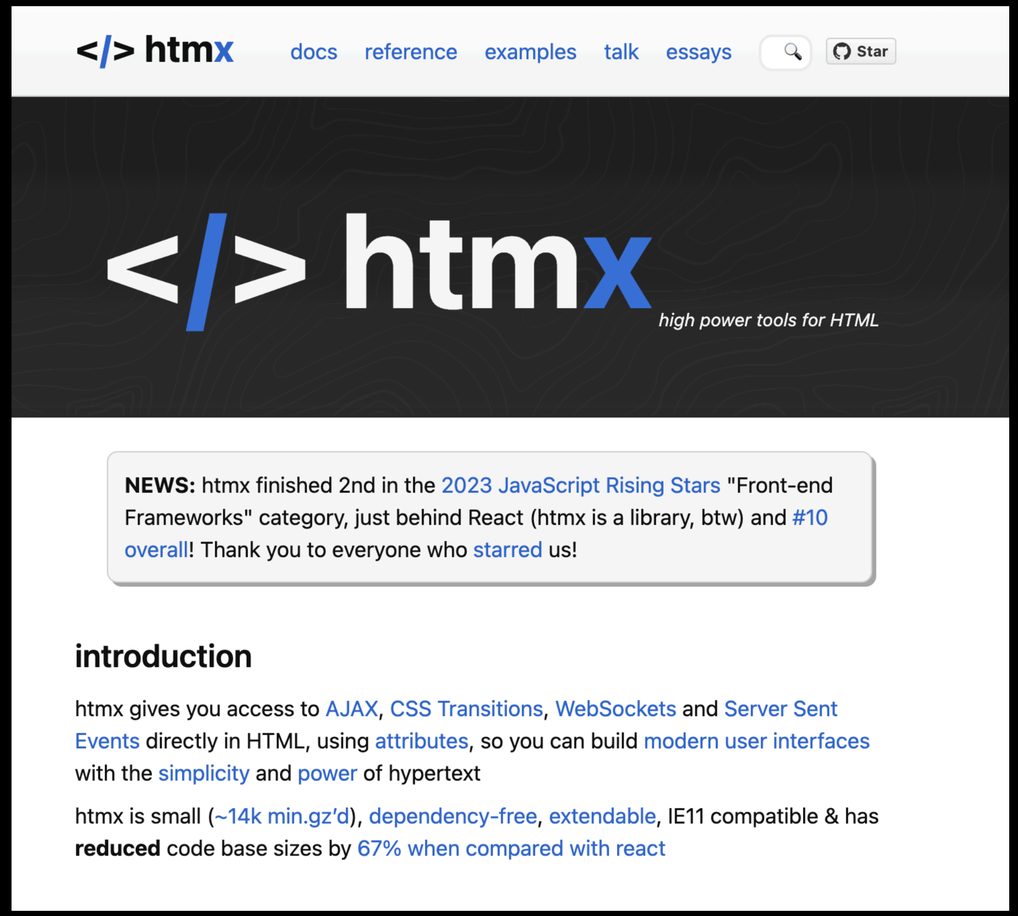
Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49 ■ 50–89 ● 90–100



METRICS — Expand view

First Contentful Paint 0.8 s	Largest Contentful Paint 0.8 s
Total Blocking Time 0 ms	Cumulative Layout Shift 0
Speed Index 0.8 s	

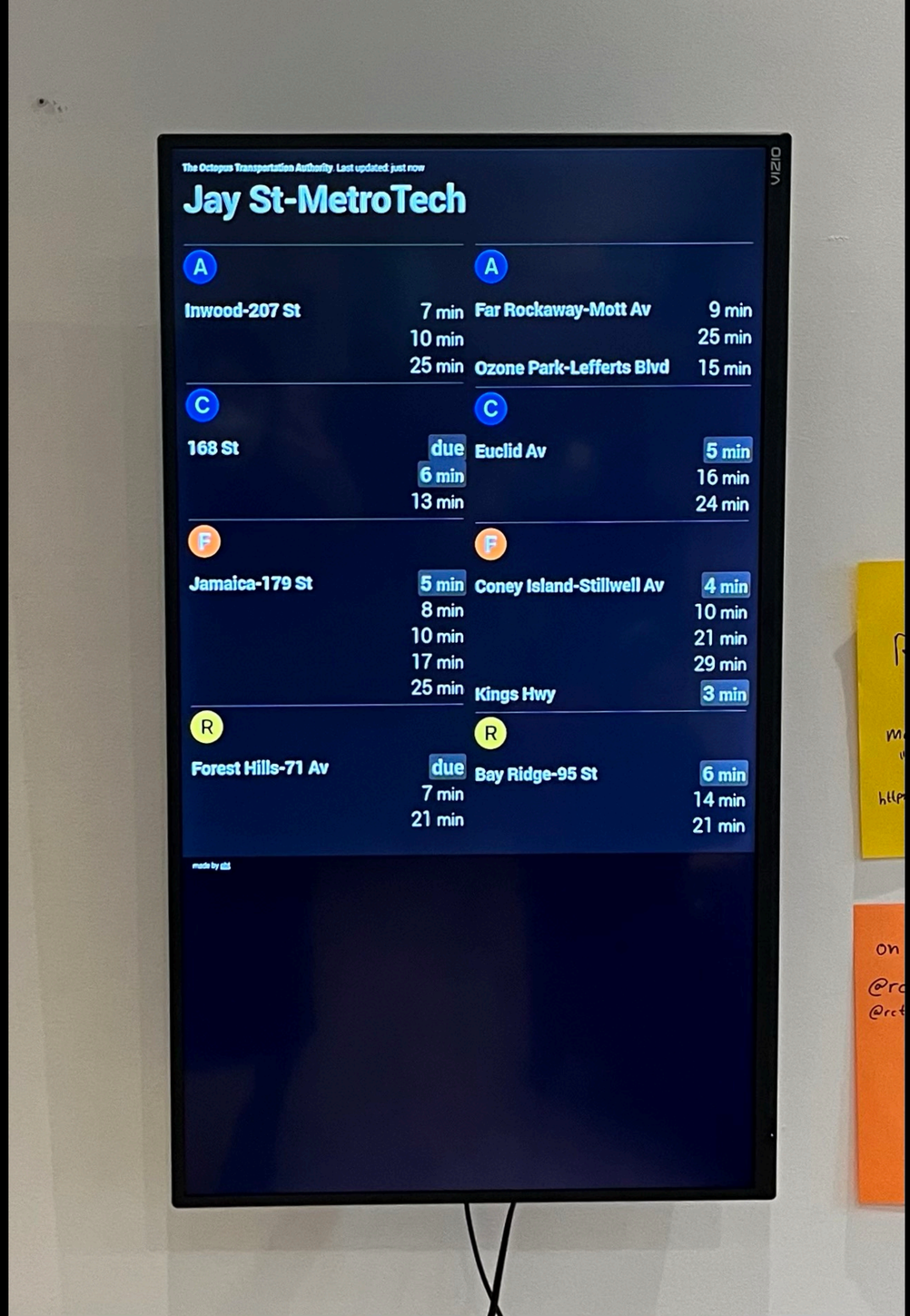

htmx
docs
reference
examples
talk
essays
Star
htmx
high power tools for HTML
NEWS: htmx finished 2nd in the 2023 JavaScript Rising Stars "Front-end Frameworks" category, just behind React (htmx is a library, btw) and #10 overall! Thank you to everyone who starred us!
introduction
htmx gives you access to AJAX, CSS Transitions, WebSockets and Server Sent Events directly in HTML, using attributes, so you can build modern user interfaces with the simplicity and power of hypertext
htmx is small (~14k min.gz'd), dependency-free, extendable, IE11 compatible & has reduced code base sizes by 67% when compared with react

htmx.org

```
</header>
<main hx-get="{{url_for('countdown', mode=mode, stop_ids=stop_ids)}}" hx-trigger="every 15s" hx-swap="morph:outerHTML"
  hx-select="main > *">
```

Future Work

1. Making it cache better
2. Adding a UI to navigate stations? Trains?
3. Adding support for Alerts
4. Schedules
5. **What do you want to see?**



help me with...

1. Ideas
2. Pairing

tysm <3

Matthew Long (W'14)
For giving me a primer to the MTA data

Countless, countless people I've talked to in the hub
If you'd like to talk to me, say hi :)

shoutout to:

https://wheresthefuckingtrain.com/