

Bringing ideas to life with IoT

Visualizing noise pollution

Ryan Carroll - Senior Software Engineer @ CoverMyMeds

A little about me

- Full-stack development for 8 years
- Outdoor lover
- Weekend audio engineer
- Father and husband



carrollmedia.dev



Where we're going

- What is IoT, anyways?
- Why noise?
- High-level project walkthrough
- Tech deep dive
- So, what were the results? *Demo time!*
- What does this mean for you?



What is IoT?



A thing with internet!

What is noise pollution?

"any unwanted or disturbing sound

that **affects the health and well-being**

of humans and other organisms."

- National Geographic



What is noise pollution?

“...To be precise, noise becomes **harmful** **when it exceeds 75 decibels** (dB) and is painful above 120 dB...noise levels [should] be kept below 65 dB during the day and indicates that restful **sleep is impossible** with nighttime ambient noise levels **in excess of 30 dB.**”

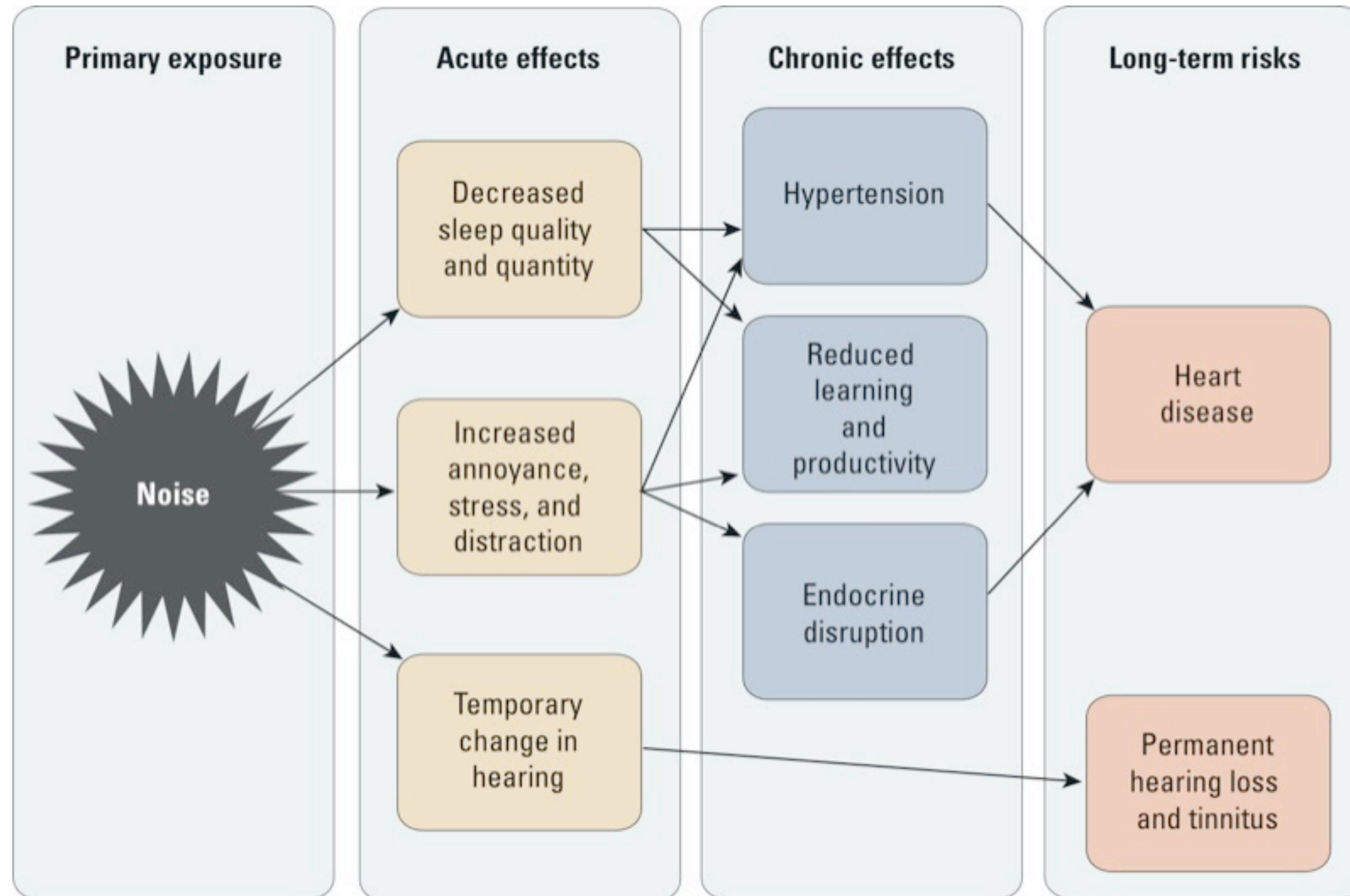
Source: iberdrola.com



Sources include:



Why is it a big deal?



Source: nih.gov

Figure 11. Key impacts of exposure to unhealthy noise levels, based on END thresholds, in the EU in 2017



*Disability adjusted life years

Source: [EEA \(2022c\)](#).

How loud is too loud?

Source: hearinghealthfoundation.org

NOISE LEVELS

Sounds at or below 70 dB are safe.



Leaves
rustling /
Whisper

Background
music

20 30 30 – 50 60 70

Ticking
watch

Average
room
noise

Average
office
noise

Sounds above 70 dB are harmful.

Landscaping
equipment
(from inside
a house)

City traffic
(from inside
a car) / Noisy
restaurant

Food processor
/ DJ'd school
dance / Crowing
rooster

Motorcycle /
Automatic
hand dryer

Trombone /
Dog barking
in ear / Ice
cream truck

Jack-
hammer

Gunshot

75 80 85 90 95 95 – 100 100 105 – 110 110 110 – 120 130 135 140

Inside an
airplane /
Electric
vacuum

Hairdryer

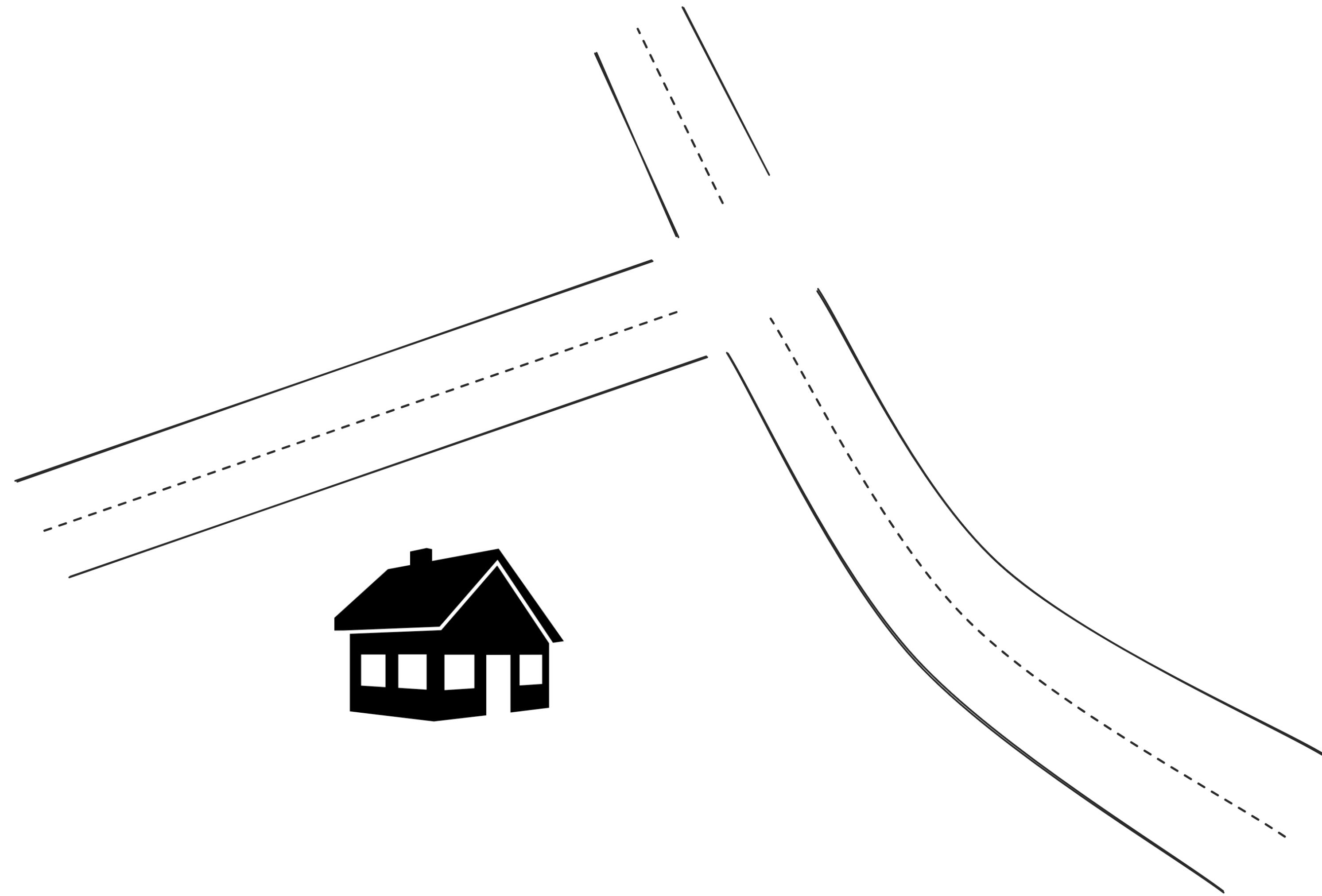
Approaching
subway train /
Car horn at
16 feet / Pro
sports games

Nightclubs
and bars /
Gas-powered
leaf blower /
Ice cream
truck

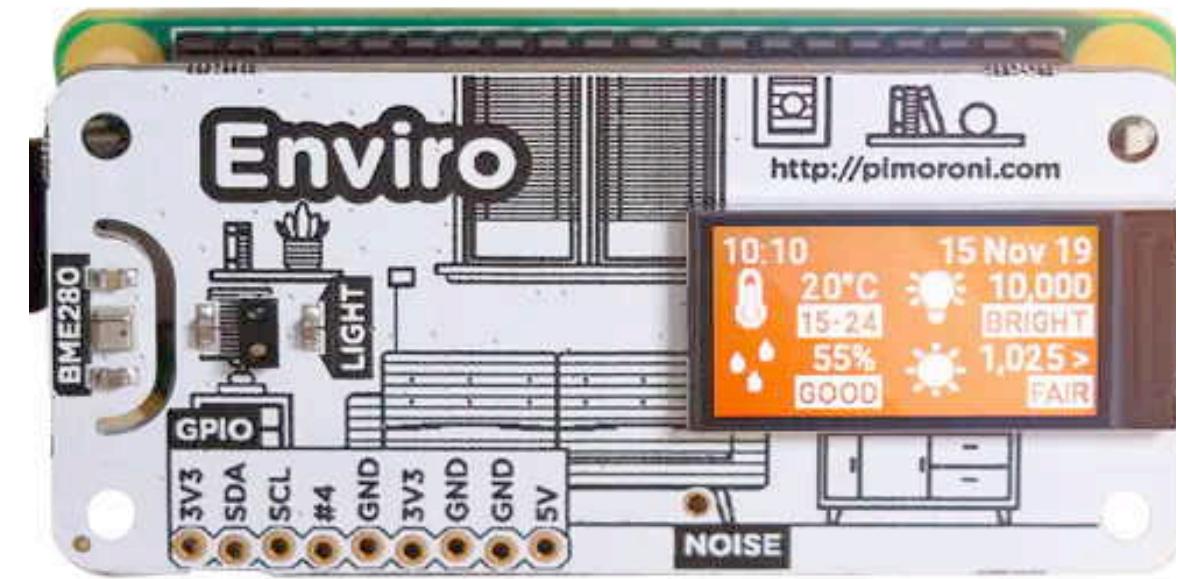
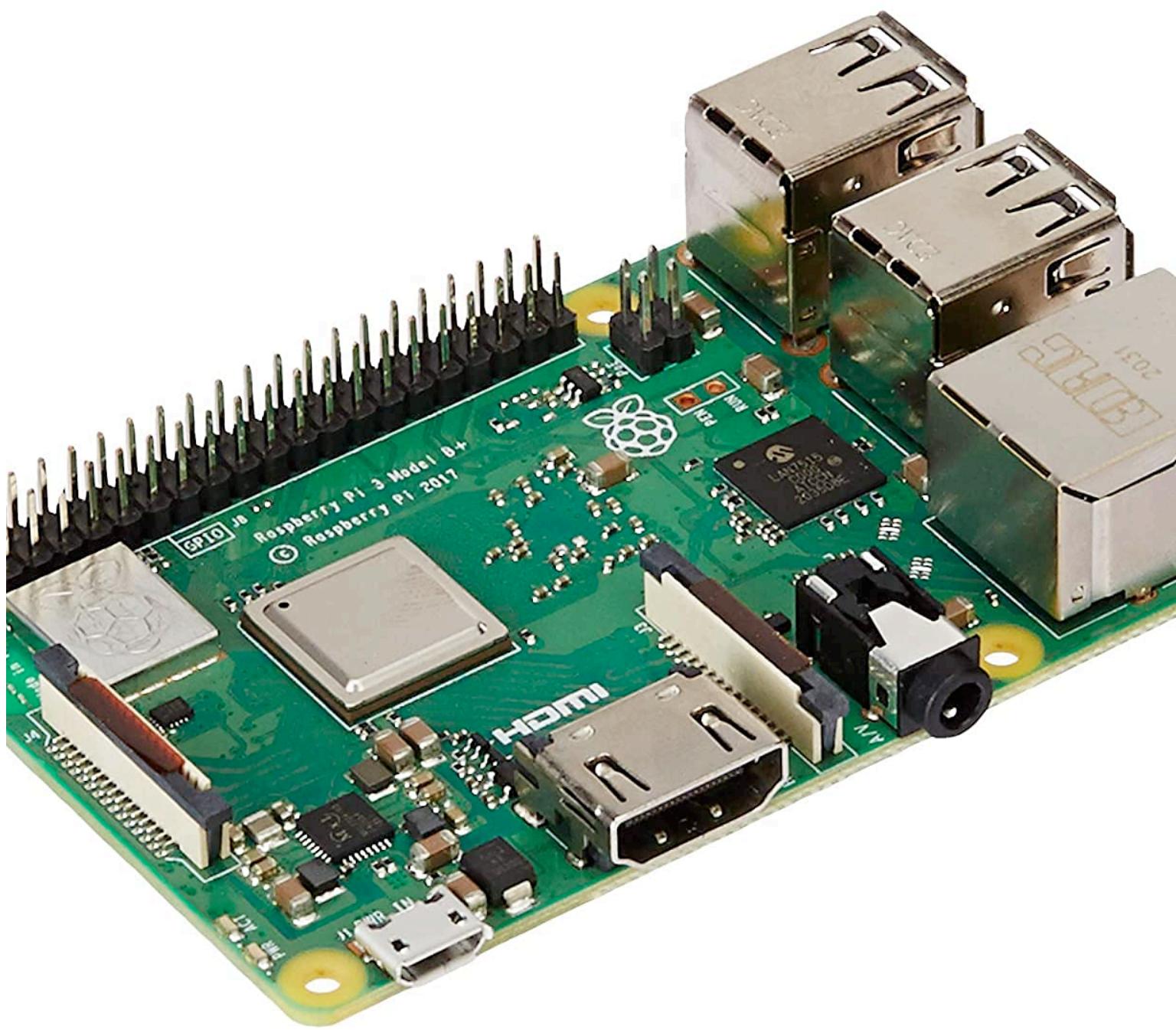
Rock or
pop concert /
Siren

Jet engine
from 100
yards

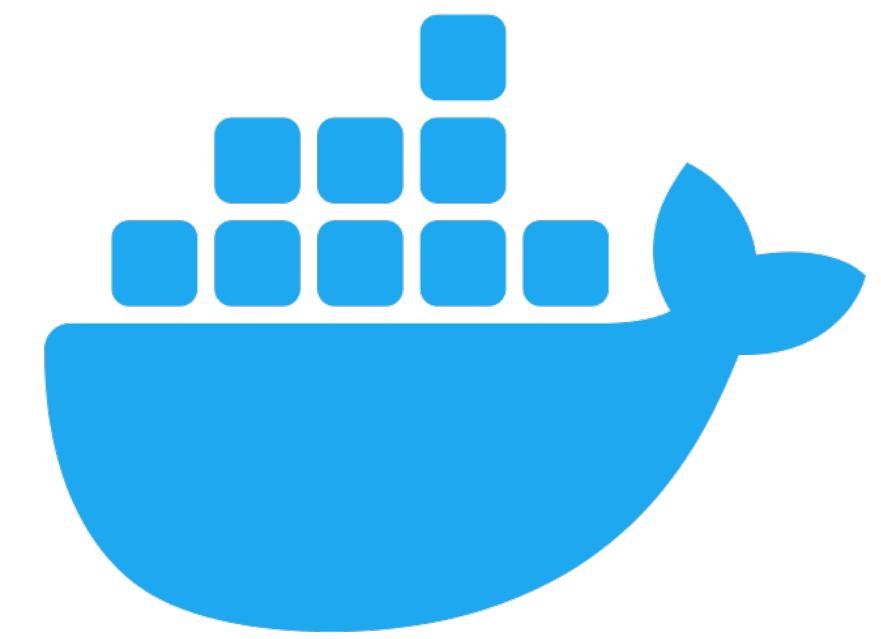
Ok, so why did I choose noise?



The project: hardware



The project: software



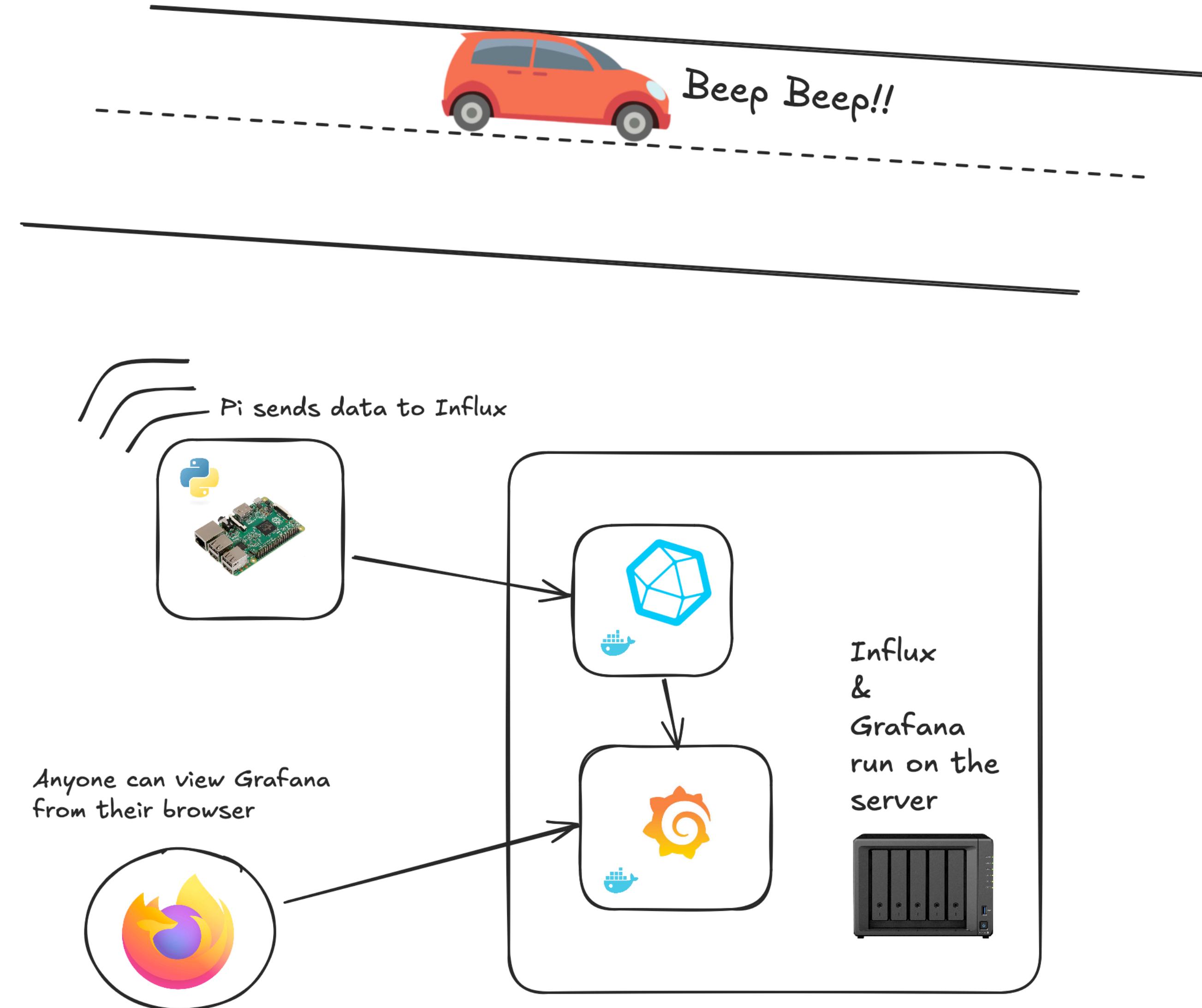
Python

InfluxDB

Grafana

Docker

Bringing it all together



The script

- Reads decibel level from the mic
- Send the data to InfluxDB
- Wait...and loop
- Runs in a docker container

```
def log_db_to_influx(self, db_reading):
    """Logging decibel readings to influx DB"""

    bucket = "environmental"
    org = "carrollmedia"
    token = os.environ['API_TOKEN']
    url="http://192.168.8.3:8086"

    client = influxdb_client.InfluxDBClient(
        url=url,
        token=token,
        org=org
    )

    write_api = client.write_api(write_options=SYNCHRONOUS)

    p = influxdb_client.Point("environment") \
        .tag("location", "Central College") \
        .field("dB", db_reading)

    write_api.write(bucket=bucket, org=org, record=p)
```

Why InfluxDB

- Time-series database
- Client libraries
- Integrations with Grafana
- Community support

The screenshot shows the InfluxDB Data Explorer interface. On the left is a dark sidebar with white icons: a globe, a blue square with a 'c', an upward arrow, a left arrow, a pen, a grid, a calendar, a bell, and a gear. To the right is a main area with a header "Data Explorer". Below the header are two buttons: "Graph" with a dropdown arrow and "CUSTOMIZE" with a gear icon. The main area contains several panels:

- A large empty dark box.
- A "Query 1" panel with a "+" button.
- A "FROM" panel with a search bar labeled "Search for a bucket" and a list of buckets: "environmental", "environmental-downsampled", and "read_noise" (which is highlighted in blue).
- A panel on the far right showing the message "No tag keys found".

InfluxDB - the how

- Install via Docker compose
- Create a bucket
- Generate an API key
- Install the Influx Python library
- Send readings to the bucket

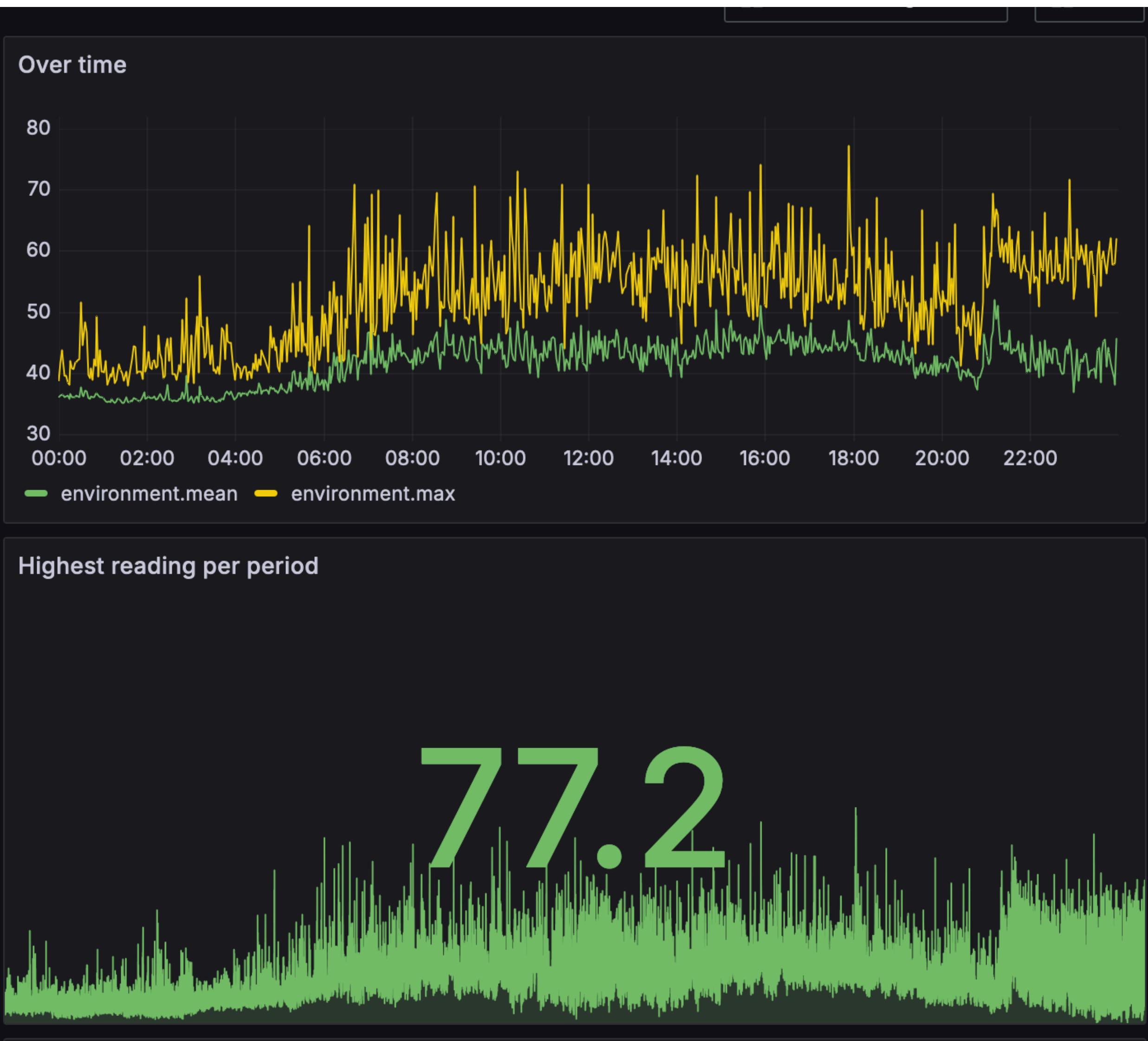
```
version: '3'

services:
  influxdb:
    image: influxdb:2.6-alpine
    env_file:
      - influx2.env
    volumes:
      # Mount for influxdb data directory and configuration
      - influxdbv2:/var/lib/influxdb2:rw
    ports:
      - "8086:8086"
  telegraf:
    image: telegraf:1.25-alpine
    depends_on:
      - influxdb
    volumes:
      # Mount for telegraf config
      - ${PWD}/telegraf/mytelegraf.conf:/etc/telegraf/telegraf.conf:ro
    env_file:
      - influx2.env

  volumes:
    influxdbv2:
```

Why Grafana

- Easy to use
- Built for observability and metrics
- Datavis library
- Open source

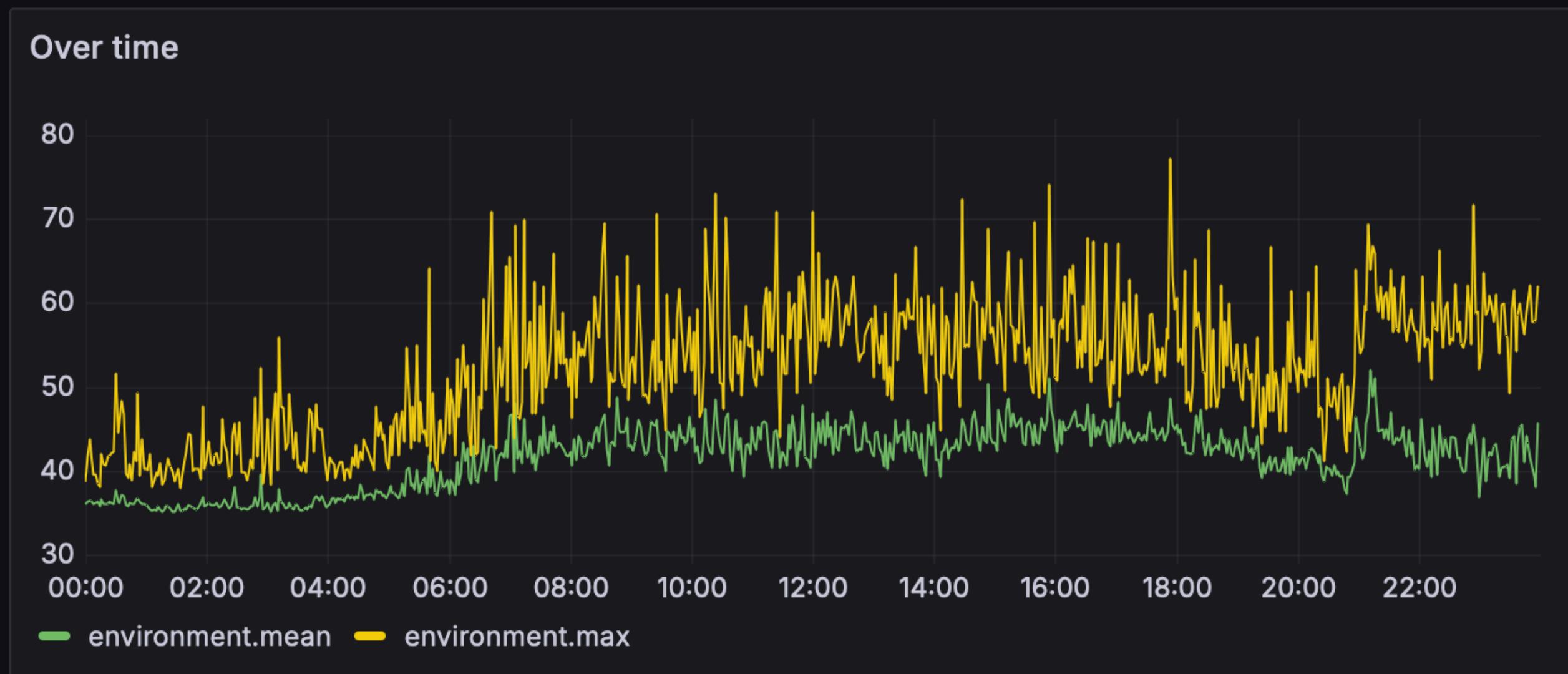
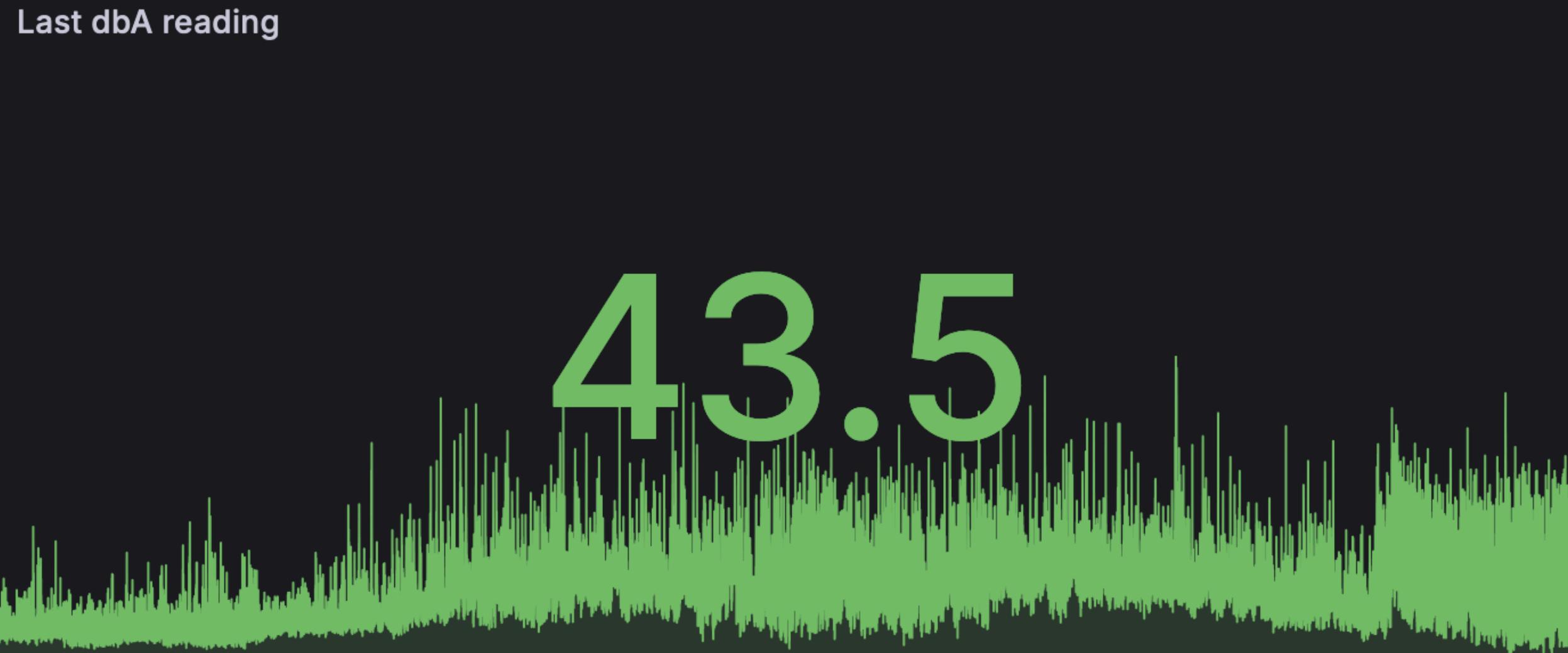


Grafana - the how

- Install via Docker compose
- Configure InfluxDB connector
- Create a dashboard
- Build fun charts!

```
version: '3'
services:
  grafana:
    image: grafana/grafana-oss
    user: grafana:grafana_group
    container_name: grafana
    restart: unless-stopped
    environment:
      - GF_SERVER_ROOT_URL=http://data.carrollmedia.dev/
      - GF_SERVER_HTTP_ADDR=
      - GF_SERVER_HTTP_PORT=3000
      - GF_SERVER_DOMAIN=carrollmedia.dev
      - GF_SERVER_CERT_KEY=/var/lib/certs/grafana.key
      - GF_SERVER_CERT_FILE=/var/lib/certs/grafana.crt
      - GF_SERVER_ENFORCE_DOMAIN=False
      - GF_SERVER_PROTOCOL=https
    ports:
      - '3000:3000'
    volumes:
      - grafana-storage:/var/lib/grafana
      - $PWD:/var/lib/certs
volumes:
  grafana-storage: {}
```

The results



What's next?

- Deploy to additional sites
- Build an action plan/proposal
- Train a model
 - Weather/noise correlation
 - Time/noise correlation
- ID noise type/source



What can you do?

- UV/air quality correlation
- Wildlife tracker/metrics
- Monitor PH in water supply
- Track garden soil moisture
 - Plant growth correlation



Keys to success

- Stay curious
- Pace yourself
- Be patient
- Start small

