



Start Automating InfluxDB Deployments at the Edge with balena

January 24, 2023





Jay Clifford

Developer Advocate, InfluxData

Past life: Sales Engineer for IIoT Solutions.

Passion: Autonomous and Vision based projects.

Driven: To make IoT accessible to all.

Belief: Industrial IoT's success belongs to the domain experts.



balena

| influxdata®



Marc Pous

IoT Giant

Developer Advocate, balena.io

Past life: IoT and Smart Cities during the last 15 years.

Passion: Educative IoT devices.

Driven: Make the planet a better place with IoT.

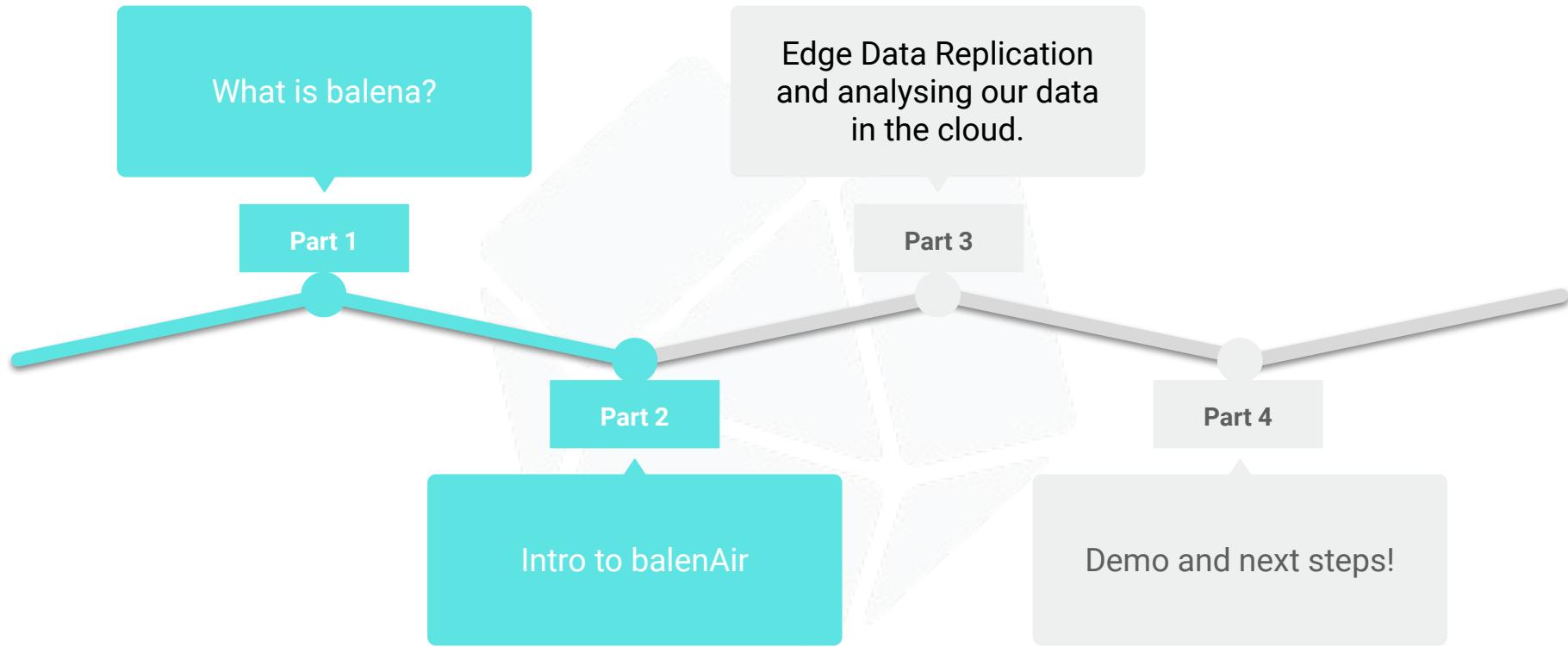
Belief: IoT success will come when everyone will collaborate.



balena

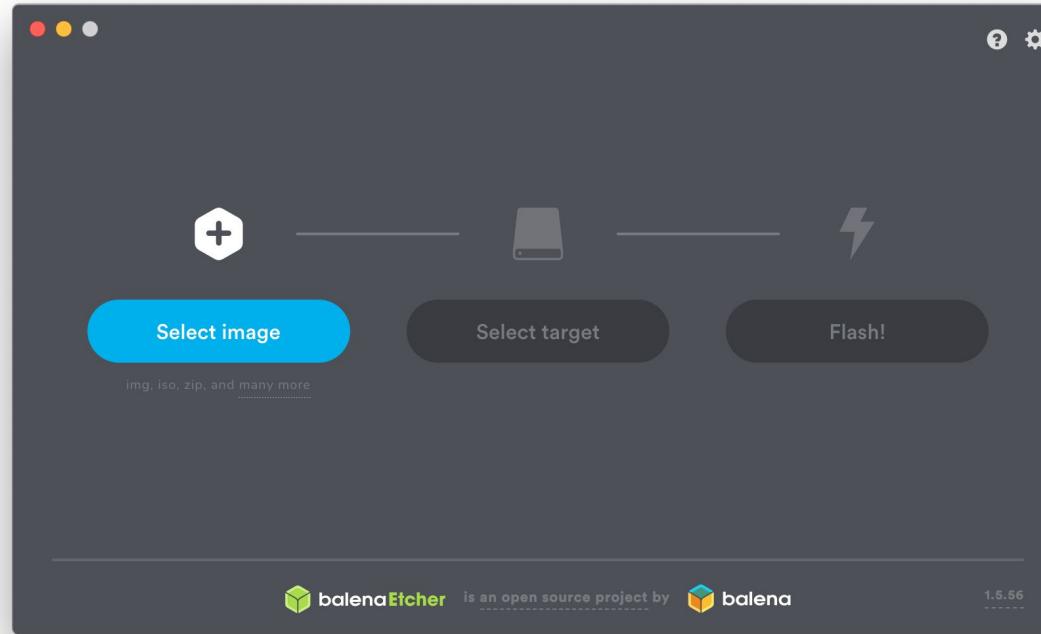


influxdata®

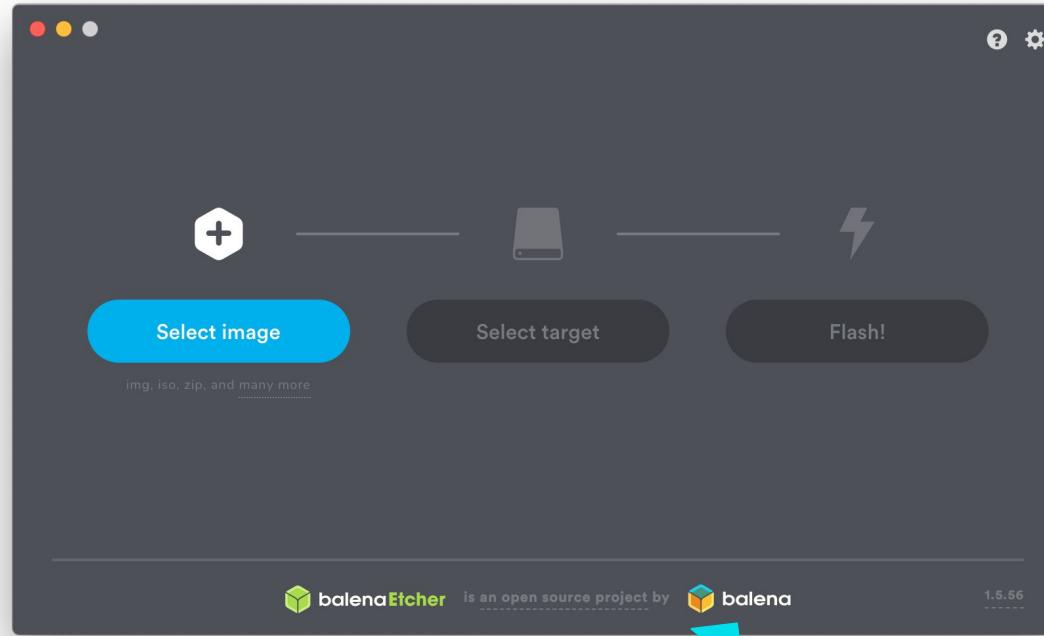


What is balena?

balena - Anyone?

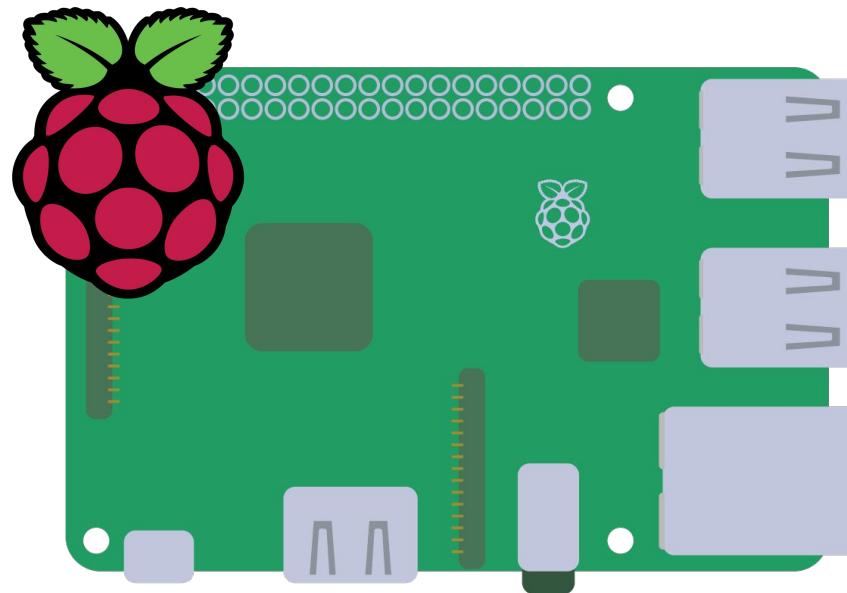


balena - Anyone?

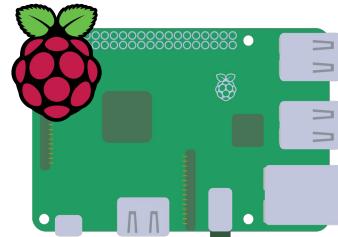
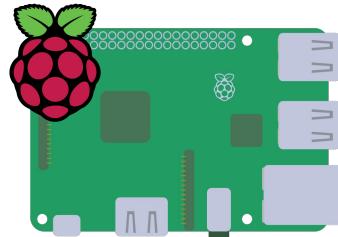
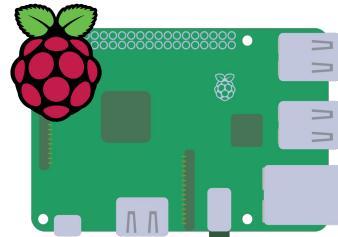
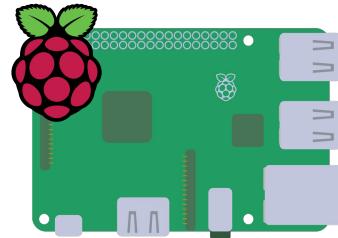
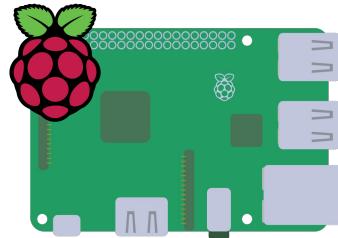
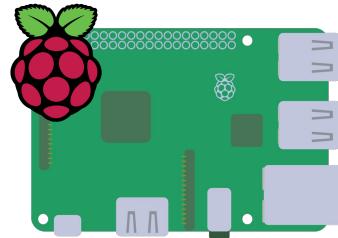
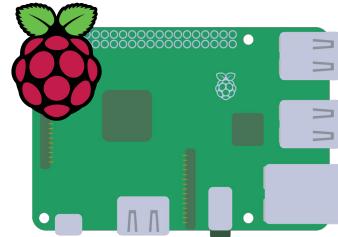
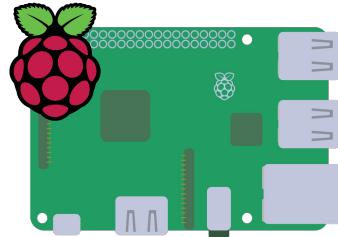
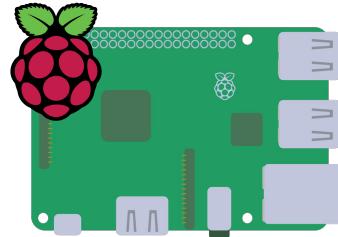


“Platform built to develop, deploy, and manage
fleets of connected devices at scale.”

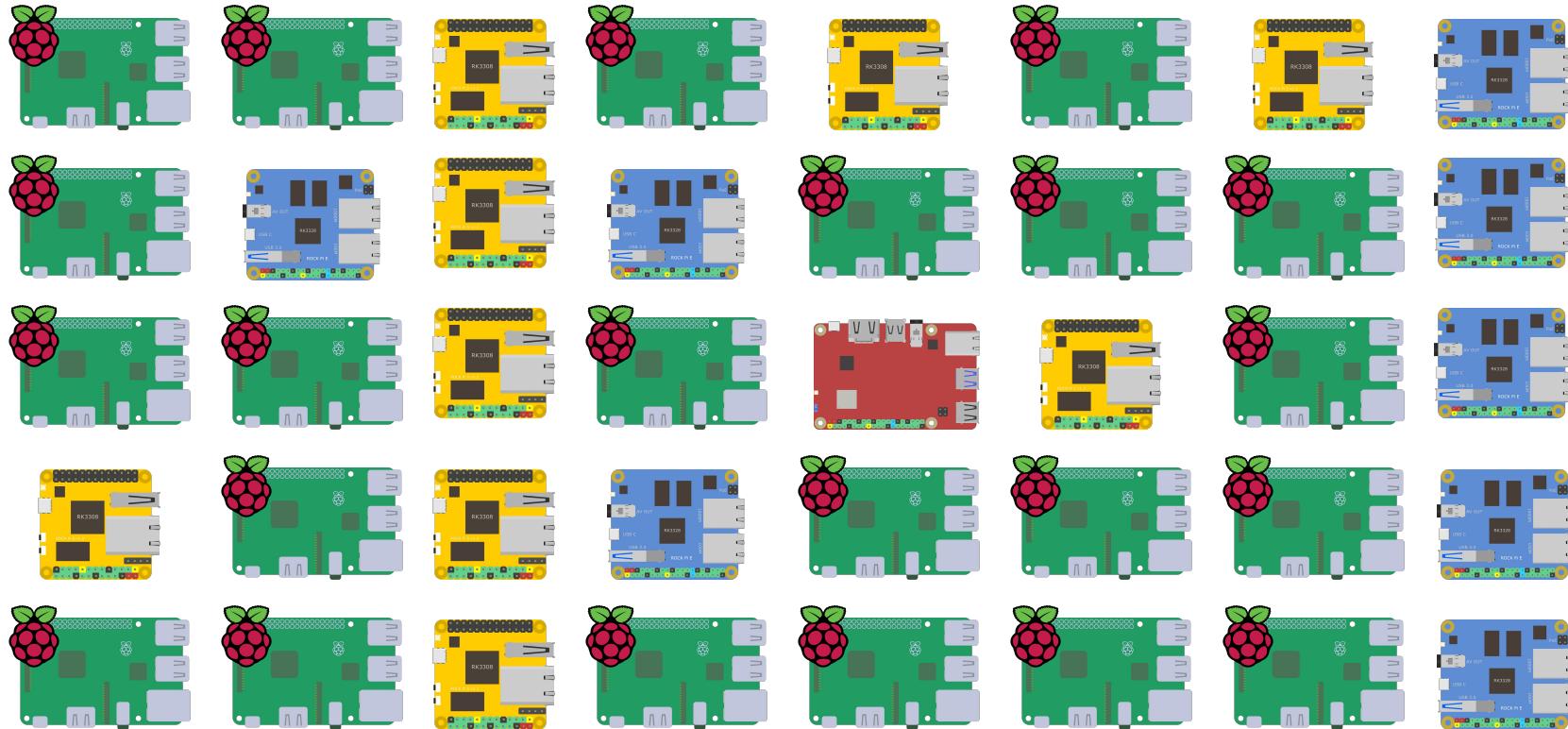
balena - Fleet Management



balena - Fleet Management



balena - Fleet Management



balena - At a Glance

The screenshot displays the balenaCloud web interface with a dark theme. The top navigation bar includes links for "Getting Started", "Docs", "Forums", "Status", and "Jon Rich". A user profile icon is also present.

Left Sidebar: Shows a tree view of organization structures: Organizations, balenAir, Fleets, balenAir, Devices, and the selected device **iaq-alan-full**. Below this are links for "Summary", "Device Variables", "Device Configuration", "Actions", "Settings", "Diagnostics", and "Location". A "Changelog" section at the bottom indicates version v15.15.20.

Device Overview: The main panel shows details for the device **iaq-alan-full**. Key information includes:

- STATUS:** Online (green checkmark)
- UUID:** 4491931
- TYPE:** Raspberry Pi 3
- ONLINE FOR:** 1 day
- HOST OS VERSION:** balenaOS 2.95.3+rev1
- OS VARIANT:** production
- SUPERVISOR VERSION:** 12.11.38
- CURRENT RELEASE:** e392582 (green checkmark)
- TARGET RELEASE:** e392582 (blue question mark)
- SUPPORT ACCESS:** Off
- FLEET:** balenair/balenair
- IS ACTIVE:** (green checkmark)
- LOCAL IP ADDRESS:** 192.168.1.110
- PUBLIC IP ADDRESS:** 98.114.64.104
- MAC ADDRESS:** B8:27:EB:F0:14:61
- TAGS (0):** No tags configured yet
- PUBLIC DEVICE URL:** (Off) / (On)
- NOTES:** Add device notes

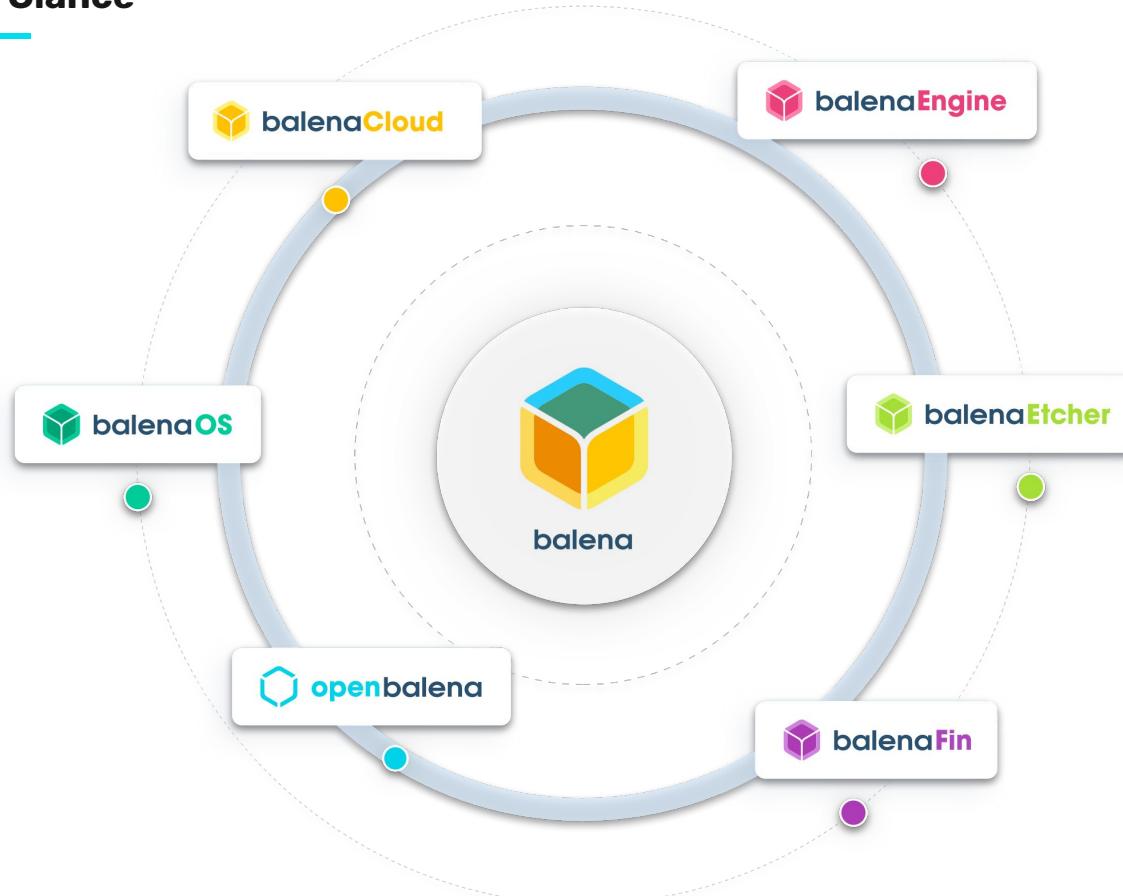
System Monitoring: A summary of system resources:

- CPU:** ~9% Usage, ~49C Temperature
- Memory:** 322 MB/473 MB
- Storage:** 1.6 GB/28.3 GB (dev/mmcblk0p6)

Logs: A log viewer showing entries from 08.11.22 14:13:15. The log includes sensor readings for temperature, humidity, VOC levels, and CO2 concentration, along with a Telegraf log entry for an influxdb write operation.

Terminal: A terminal session window where a user can select a target and start a session.

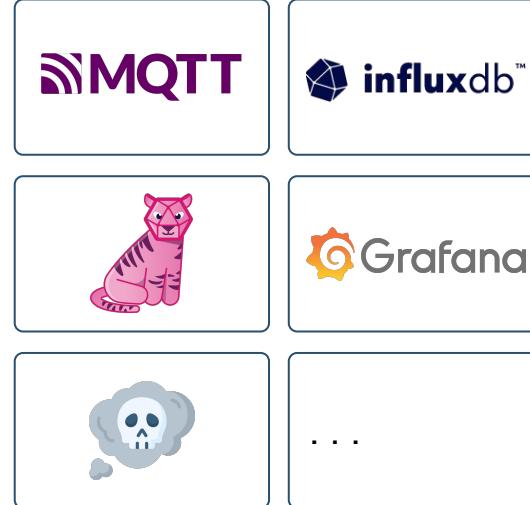
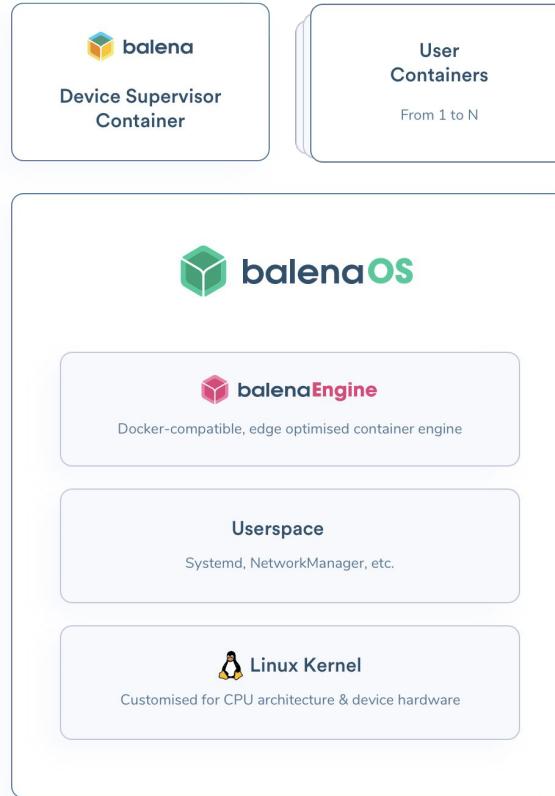
balena - At a Glance



balena - At a Glance



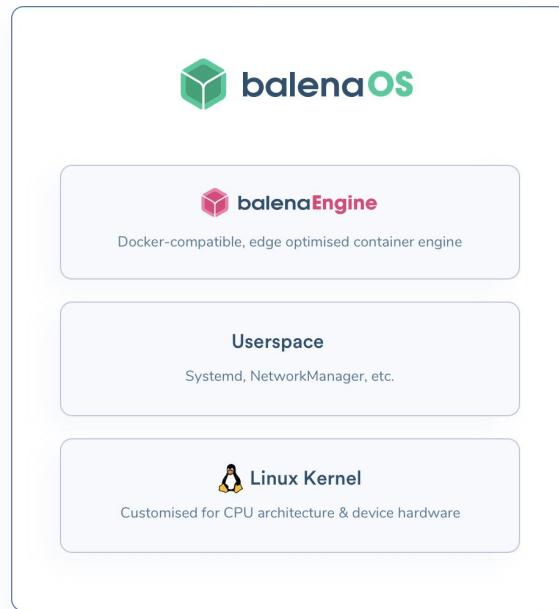
balena - At a Glance



balena - At a Glance



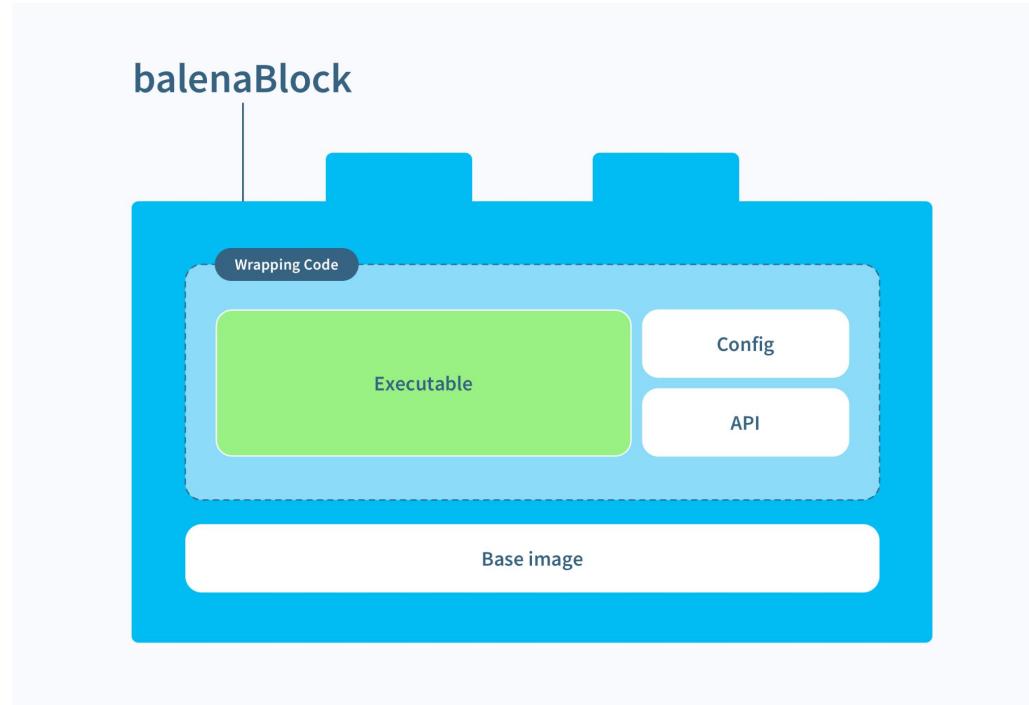
balena - At a Glance



balena - At a Glance



balena - What is a Block

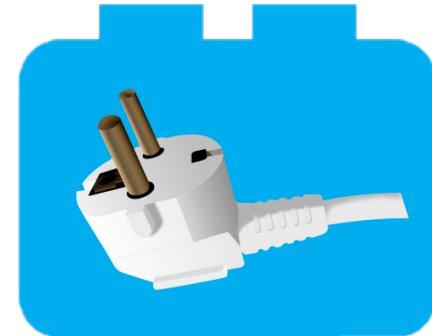


<https://blog.balena.io/introducing-balenablocks-jumpstart-your-iot-app-development/>

balena - Connector Block

The connector block wraps **Telegraf** in code which discovers and intelligently connects data sources and sinks.

For example, the connector block will find an MQTT broker running on the device and subscribe to a sensor topic. An application developer can simply send JSON-formatted data to this topic and it will be ingested by connector. Similarly, the block will find an instance of InfluxDB running on the device, and send any ingested data to it.



The use of environment variables allows a developer or fleet owner to enable lots of other data sources and sinks, both on the device and externally hosted.

<https://github.com/balena-labs-projects/connector>

[https://hub.balena.io/organizations/balenalabs\(blocks\)/connector](https://hub.balena.io/organizations/balenalabs(blocks)/connector)



balenAir

What is balenAir ?



A “build it yourself” Indoor **Air Quality Device - IAQ**.

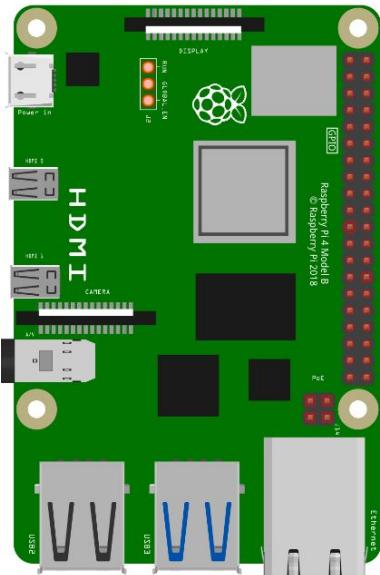
Motivation: Dedicated to making environmental information visibly more accessible.

I open source I highly customizable
I easy to use I privacy I

IAQ = (CO₂, VOC and particulate sensors)

0 (best air quality) - 99 (hazardous)

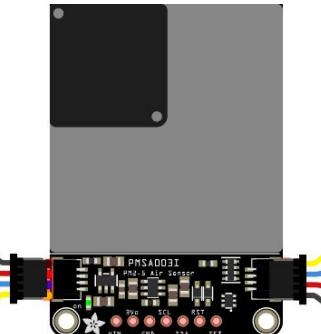
Sensors and Hardware



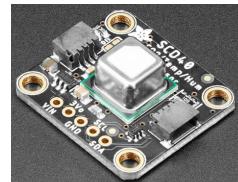
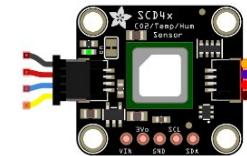
Raspberry PI 3 / 4b



SparkFun Qwiic LED Stick

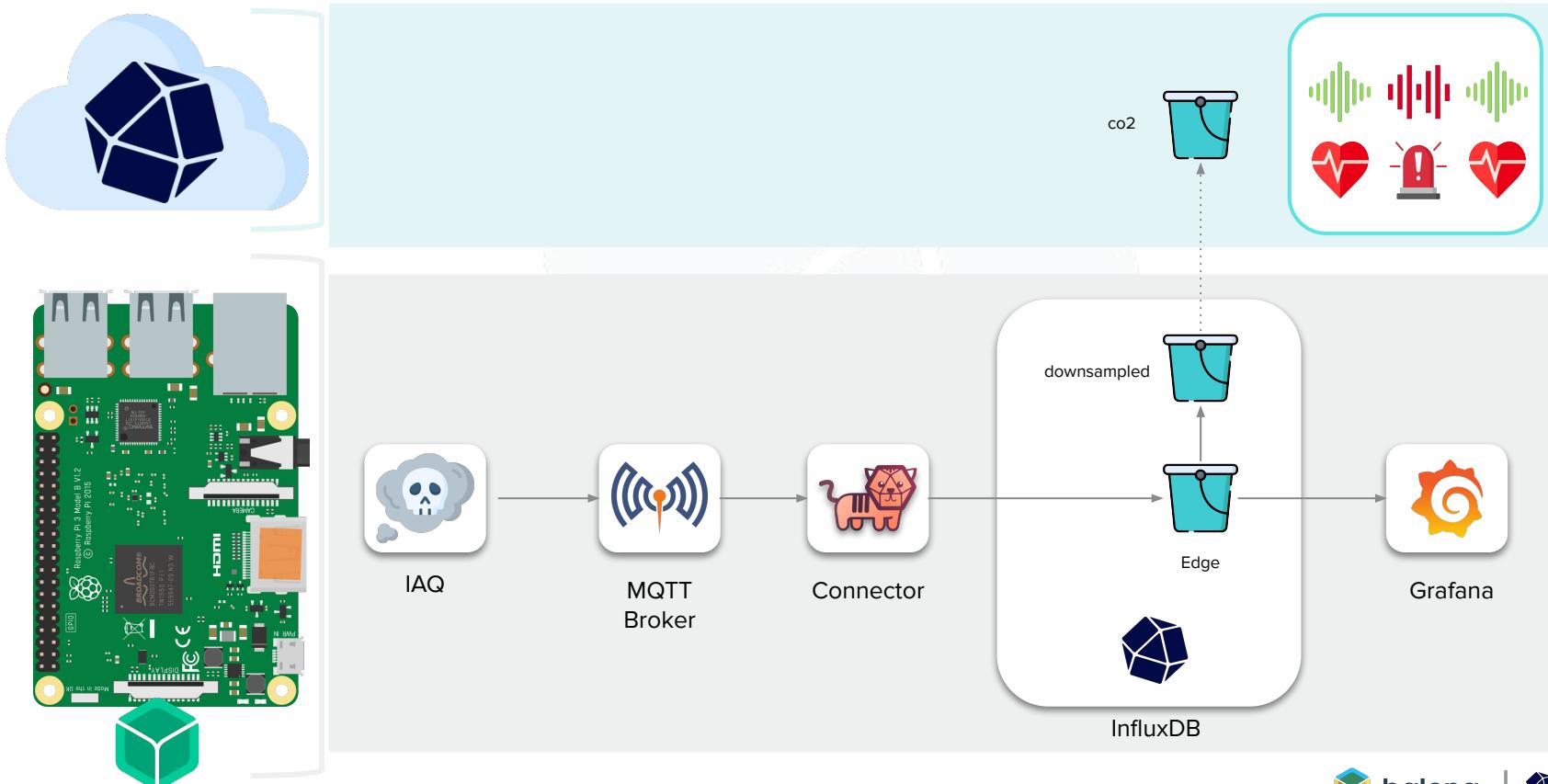


PMSA003I



SCD-40

Demo Architecture



Edge Data Replication



influxdb cloud™

Cloud

northbound



influxdb™

Write
Endpoint



Telegraf

replication queue



generator

Edge

Analyzing our data in the cloud



```

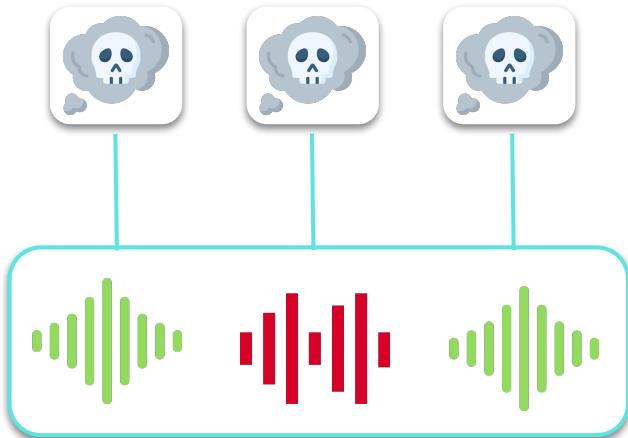
1 import "math"
2
3 bicycles3 = from(bucket: "smartcity")
4   |> range(start:2021-03-01T00:00:00Z, stop: 2021-04-01T00:00:00Z)
5   |> filter(fn: (r) => r._measurement == "city_IoT")
6   |> filter(fn: (r) => r._field == "counter")
7   |> filter(fn: (r) => r.source == "bicycle")
8   |> filter(fn: (r) => r.neighborhood_id == "3")
9   |> aggregateWindow(every: 1h, fn: mean, createEmpty:false)
10
11 bicycles4 = from(bucket: "smartcity")
12   |> range(start:2021-03-01T00:00:00Z, stop: 2021-04-01T00:00:00Z)
13   |> filter(fn: (r) => r._measurement == "city_IoT")
14   |> filter(fn: (r) => r._field == "counter")
15   |> filter(fn: (r) => r.source == "bicycle")
16   |> filter(fn: (r) => r.neighborhood_id == "4")
17   |> aggregateWindow(every: 1h, fn: mean, createEmpty:false)
18
19 join(tables: {neighborhood_3: bicycles3, neighborhood_4: bicycles4}, on: ["_time"], method: "inner")
20   |> keep(columns: ["_time", "_value_neighborhood_3","_value_neighborhood_4"])
21   |> map(fn: (r) => ({
22     r with
23       difference_value: math.abs(x: (r._value_neighborhood_3 - r._value_neighborhood_4))
24   }))

```

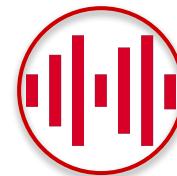
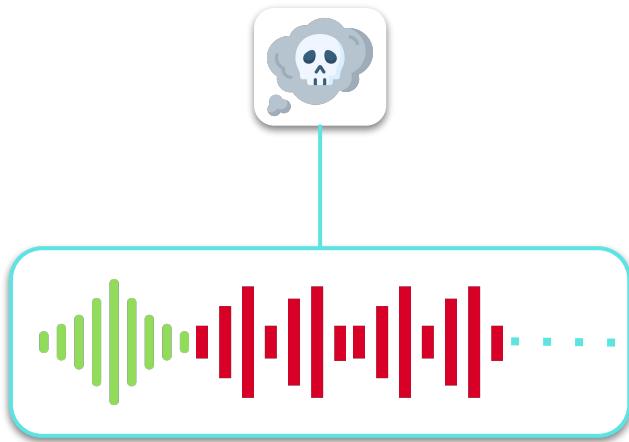
Flux

A functional language designed for
querying, analyzing, and acting on data.

Anomaly Detection



Forecasting





Let's start!

Deployment with balena

The screenshot shows the balenaHub application page for the 'balenAir_InfluxDB_V2' app. The left sidebar shows the organization 'g_jay_clifford' and the app 'balenAir_InfluxDB_V2'. The main content area displays the app's summary, usage instructions, and release notes. The summary includes a description of the app as an indoor air quality monitoring device, compatibility with Raspberry Pi and BeagleBoard, and a 'Deploy' button. The usage instructions provide steps for download and setup. The release notes show version v2.0.0 from 10 Jan 2023 with no release notes.

balenAir_InfluxDB_V2 by g_jay_clifford

What is balenaHub? Contribute Community

Back to Apps

balenAir_InfluxDB_V2 by g_jay_clifford's Organization

Description: Build an indoor air quality monitoring device featuring a simple real time display and a detailed web dashboard www.balenair.com

Works With: Raspberry Pi, BeagleBoard

Version v2.0.0 [View code](#) | 362 MB | last updated at: 10 Jan 2023

Deploy Report issue

Usage instructions

Application download: Once you power up your device you'll need to allow some time for it to download the application, this typically takes 5-10 minutes but it can take more depending on your internet connection and what device you are using.

Usage: When the download is completed the application will start. Depending on the display chosen you should see a LED test sequence and then shortly an indication of the current air quality. Please note that if this is the first time setting up the device and depending on what sensors you have installed you may need to wait 12 hours for baseline readings to be calculated. See this [documentation](#) for more information.

Release notes

v2.0.0 | 10 Jan 2023

No release notes

Changelog v4.6.12

balenaHub A project by balena.io

Twitter Facebook Instagram YouTube

Terms of use

Send feedback

Tips and Tricks

Connector block tips & tricks

The screenshot shows a GitHub repository page for 'balena-labs-projects/connector'. The 'Code' tab is selected, displaying the contents of the 'connector/plugins' directory. The commit history shows the following changes:

Commit	Description	Date
Dzeri96 Added configurable pull interval	Added ExternalHttpPull plugin	Nov 8, 2022
... (pycache)	Fixes for bugs raised during initial testing	2 years ago
applicationInsightsOutput.py	Added ExternalHttpPull plugin	2 years ago
baseConfig.py	Added configurable pull interval	2 months ago
deviceMetricsInput.py	Added temp to device metrics plugin	2 years ago
externalHttpListener.py	code cleanup	last year
externalHttpPullInput.py	Fix bugs and image build failure	last year
httpOutput.py	added Content-Type header to http json output	last year
influx2Output.py	Add influx2 autowiring script	last year
influxOutput.py	documentation and influxdb name variable change	2 years ago
internalHttpPullInput.py	Fix bugs and image build failure	last year
mqttInput.py	code cleanup	last year
mqttOutput.py	Added topic prefix	2 years ago

<https://github.com/balena-labs-projects/connector>

Favourite Python Libraries

Anomaly Detection Toolkit (ADTK)

Unsupervised and rule-based time series anomaly detection

The ADTK package allows you to easily build an effective detection model from a variety or rule-based anomaly detection methods.

ARUNDO

FB Prophet

“Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects.”



Neural Prophet

A Neural Network based Time-Series model, inspired by Facebook Prophet and AR-Net (Autoregressive neural network), built on PyTorch.

Neural Prophet

Kats

“A new library to analyze time series data. Kats is a lightweight, easy-to-use, and generalizable framework for generic time series analysis”



What next?

Balena Hub

The screenshot shows the balenaHub interface. On the left, a sidebar lists 'Organizations' (g_jay_clifford), 'Apps' (balenAir_InfluxDB_V2), and navigation links 'Summary' and 'Releases'. The main content area displays the app 'balenAir_InfluxDB_V2' by g_jay_clifford's Organization. It includes a description: 'Build an indoor air quality monitoring device featuring a simple real time display and a detailed web dashboard www.balenair.com'. Below the description are two small icons of a Raspberry Pi and a sensor. To the right, there is a 'Works With' section showing compatibility with a Raspberry Pi 4 and a Raspberry Pi 3. At the top right are sharing options and a 'Deploy' button. Below the main card, under 'Usage instructions', is a 'Application download' section with a note about download time and usage instructions. Under 'Release notes', it shows version v2.0.1 from 16 Jan 2023 with no release notes.

https://hub.balena.io/organizations/g_jay_clifford/apps/balenAir_InfluxDB_V2

<https://forums.balena.io>

Learning and Community

INFLUXDB UNIVERSITY

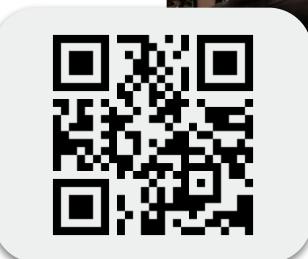
Courses | What do you want to learn? 

Login | Get InfluxDB | [Sign Up](#)

Welcome, let's begin!

Our catalog of free trainings helps you gain skills and get started quickly.

[Explore](#)

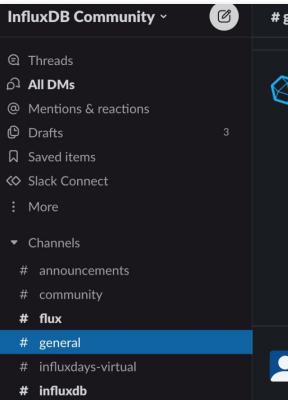


InfluxDB Community

- Threads
- All DMs
- Mentions & reactions
- Drafts
- Saved items
- Slack Connect
- More

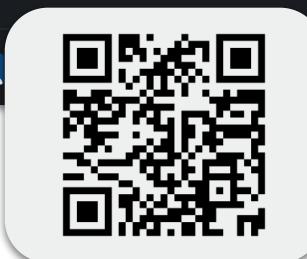
Channels

- # announcements
- # community
- # flux
- # general
- # influxdays-virtual
- # influxdb



InfluxData Blog APP 3:14 PM
Using the New Flux "types" Package - <https://www.influxdata.com/blog/using-new-flux-types-package/> (edited)

InfluxData
Using the New Flux "types" Package | InfluxData
The Flux "types" package introduces the `iType` function, which makes filtering on column types much easier.
#influxdb
Written by Sean Brickley Est. reading time 3 minutes
Mar 7th



<https://influxdbu.com/>

<https://influxcommunity.slack.com/>

Any Questions?



www.influxdata.com

Step-by-step tutorial

Deployment with balena

The screenshot shows the balenaHub application page for the 'balenAir_InfluxDB_V2' app. The left sidebar shows the organization 'g_jay_clifford' and the app 'balenAir_InfluxDB_V2'. The main content area displays the app's summary, usage instructions, and release notes. The summary includes a description of the app as an indoor air quality monitoring device, compatibility with Raspberry Pi and BeagleBoard, and a 'Deploy' button. The usage instructions provide steps for download and setup. The release notes show version v2.0.0 from 10 Jan 2023 with no release notes.

balenAir_InfluxDB_V2 by g_jay_clifford

What is balenaHub? Contribute Community

Back to Apps

balenAir_InfluxDB_V2 by g_jay_clifford's Organization

Description: Build an indoor air quality monitoring device featuring a simple real time display and a detailed web dashboard www.balenair.com

Works With: Raspberry Pi, BeagleBoard

Version v2.0.0 [View code](#) | 362 MB | last updated at: 10 Jan 2023

Deploy Report issue

Usage instructions

Application download

Once you power up your device you'll need to allow some time for it to download the application, this typically takes 5-10 minutes but it can take more depending on your internet connection and what device you are using.

Usage

When the download is completed the application will start. Depending on the display chosen you should see a LED test sequence and then shortly an indication of the current air quality. Please note that if this is the first time setting up the device and depending on what sensors you have installed you may need to wait 12 hours for baseline readings to be calculated. See this [documentation](#) for more information.

Release notes

v2.0.0 | 10 Jan 2023

No release notes

Changelog v4.6.12

balenaHub A project by [balena.io](#)

Twitter Facebook Instagram YouTube

Terms of use

Send feedback

Deployment with balena

The screenshot shows a GitHub repository page for `InfluxCommunity/balenair`. The repository is public and forked from `balenair/balenair`. It has 9 commits ahead of the `balenair:master` branch. The repository contains several files and folders, including `dashboard`, `docs`, `iaq`, `images`, `influx`, `stl`, `.gitignore`, `CONTRIBUTING.md`, `README.md`, `VERSION`, `balenair.yml`, `docker-compose.yml`, and `logo.png`. The last commit was made by `Jayclifford345` on `2266fa3` last week, with 286 commits. A prominent blue button at the bottom of the repository page says **Deploy with balena**.

This branch is 9 commits ahead of balenair:master.

File/Folder	Description	Last Commit
<code>dashboard</code>	fixed replication and dashboard issue	last week
<code>docs</code>	fixed replication and dashboard issue	last week
<code>iaq</code>	Update iaq.py	6 months ago
<code>images</code>	Added montage with white background	8 months ago
<code>influx</code>	fixed replication and dashboard issue	last week
<code>stl</code>	Update front to strengthen posts	8 months ago
<code>.gitignore</code>	added replication	last week
<code>CONTRIBUTING.md</code>	Create CONTRIBUTING.md	8 months ago
<code>README.md</code>	Update README.md	8 months ago
<code>VERSION</code>	Logo Update	6 months ago
<code>balenair.yml</code>	updated url	last week
<code>docker-compose.yml</code>	added replication	last week
<code>logo.png</code>	Logo Update	8 months ago

About
Build an indoor air quality monitoring device featuring a simple real time display and a detailed web dashboard

Code

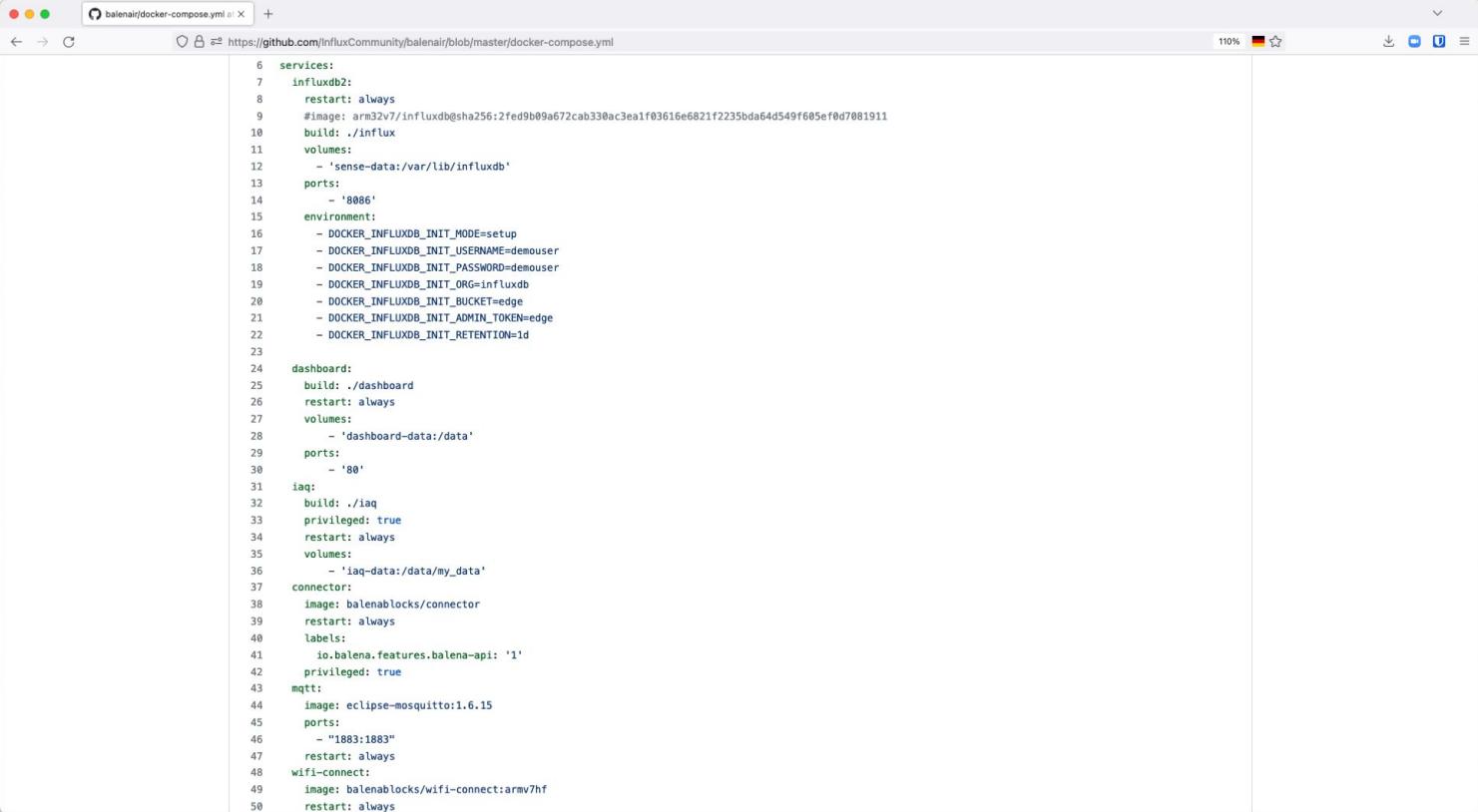
Readme
0 stars
0 watching
3 forks

Releases
No releases published

Packages
No packages published

Languages
Python 93.9%
Shell 5.6%
Dockerfile 0.5%

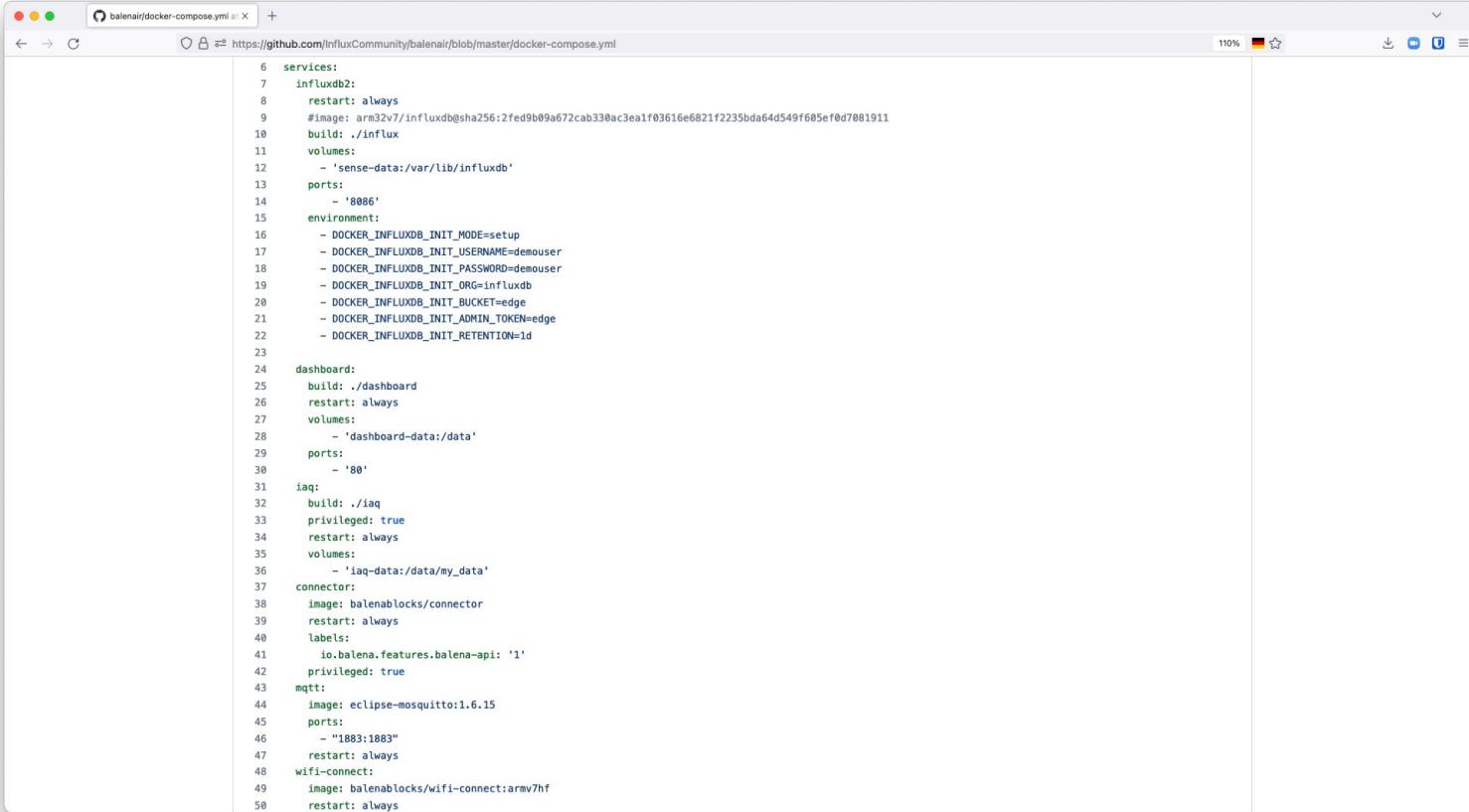
Deployment with balena



The screenshot shows a web browser window displaying the Docker Compose YAML file for the `balenair` project. The URL in the address bar is <https://github.com/influxCommunity/balenair/blob/master/docker-compose.yml>. The file contains configuration for several services: `influxdb2`, `dashboard`, `iaq`, `connector`, and `mqtt`. Each service includes details like image, ports, volumes, environment variables, and restart policies.

```
6  services:
7    influxdb2:
8      restart: always
9      #image: arm32v7/influxdb@sha256:2fed9b09a672cab330ac3ea1f03616e6821f2235bda64d549f605ef0d7081911
10     build: ./influx
11     volumes:
12       - 'sense-data:/var/lib/influxdb'
13     ports:
14       - '8086'
15     environment:
16       - DOCKER_INFLUXDB_INIT_MODE=setup
17       - DOCKER_INFLUXDB_INIT_USERNAME=demouser
18       - DOCKER_INFLUXDB_INIT_PASSWORD=demouser
19       - DOCKER_INFLUXDB_INIT_ORG=influxdb
20       - DOCKER_INFLUXDB_INIT_BUCKET=edge
21       - DOCKER_INFLUXDB_INIT_ADMIN_TOKEN=edge
22       - DOCKER_INFLUXDB_INIT_RETENTION=id
23
24   dashboard:
25     build: ./dashboard
26     restart: always
27     volumes:
28       - 'dashboard-data:/data'
29     ports:
30       - '80'
31   iaq:
32     build: ./iaq
33     privileged: true
34     restart: always
35     volumes:
36       - 'iaq-data:/data/my_data'
37   connector:
38     image: balenablocks/connector
39     restart: always
40     labels:
41       io.balena.features.balena-api: '1'
42     privileged: true
43   mqtt:
44     image: eclipse-mosquitto:1.6.15
45     ports:
46       - "1883:1883"
47     restart: always
48     wifi-connect:
49       image: balenablocks/wifi-connect:armv7hf
50     restart: always
```

Deployment with balena → flux



The screenshot shows a web browser window displaying a Docker Compose YAML file. The URL in the address bar is <https://github.com/influxCommunity/balenair/blob/master/docker-compose.yml>. The file content is as follows:

```
6 services:
7   influxdb2:
8     restart: always
9     #image: arm32v7/influxdb@sha256:2fed9b09a672cab330ac3ea1f03616e6821f2235bda64d549f605ef0d7081911
10    build: ./influx
11    volumes:
12      - 'sense-data:/var/lib/influxdb'
13    ports:
14      - '8086'
15    environment:
16      - DOCKER_INFLUXDB_INIT_MODE=setup
17      - DOCKER_INFLUXDB_INIT_USERNAME=demouser
18      - DOCKER_INFLUXDB_INIT_PASSWORD=demouser
19      - DOCKER_INFLUXDB_INIT_ORG=influxdb
20      - DOCKER_INFLUXDB_INIT_BUCKET=edge
21      - DOCKER_INFLUXDB_INIT_ADMIN_TOKEN=edge
22      - DOCKER_INFLUXDB_INIT_RETENTION=id
23
24 dashboard:
25   build: ./dashboard
26   restart: always
27   volumes:
28     - 'dashboard-data:/data'
29   ports:
30     - '80'
31   iaq:
32     build: ./iaq
33     privileged: true
34     restart: always
35     volumes:
36       - 'iaq-data:/data/my_data'
37 connector:
38   image: balenablocks/connector
39   restart: always
40   labels:
41     io.balena.features.balena-api: '1'
42   privileged: true
43 mqtt:
44   image: eclipse-mosquitto:1.6.15
45   ports:
46     - "1883:1883"
47   restart: always
48   wifi-connect:
49     image: balenablocks/wifi-connect:armv7hf
50   restart: always
```

Deployment with balena



Deployment with balena

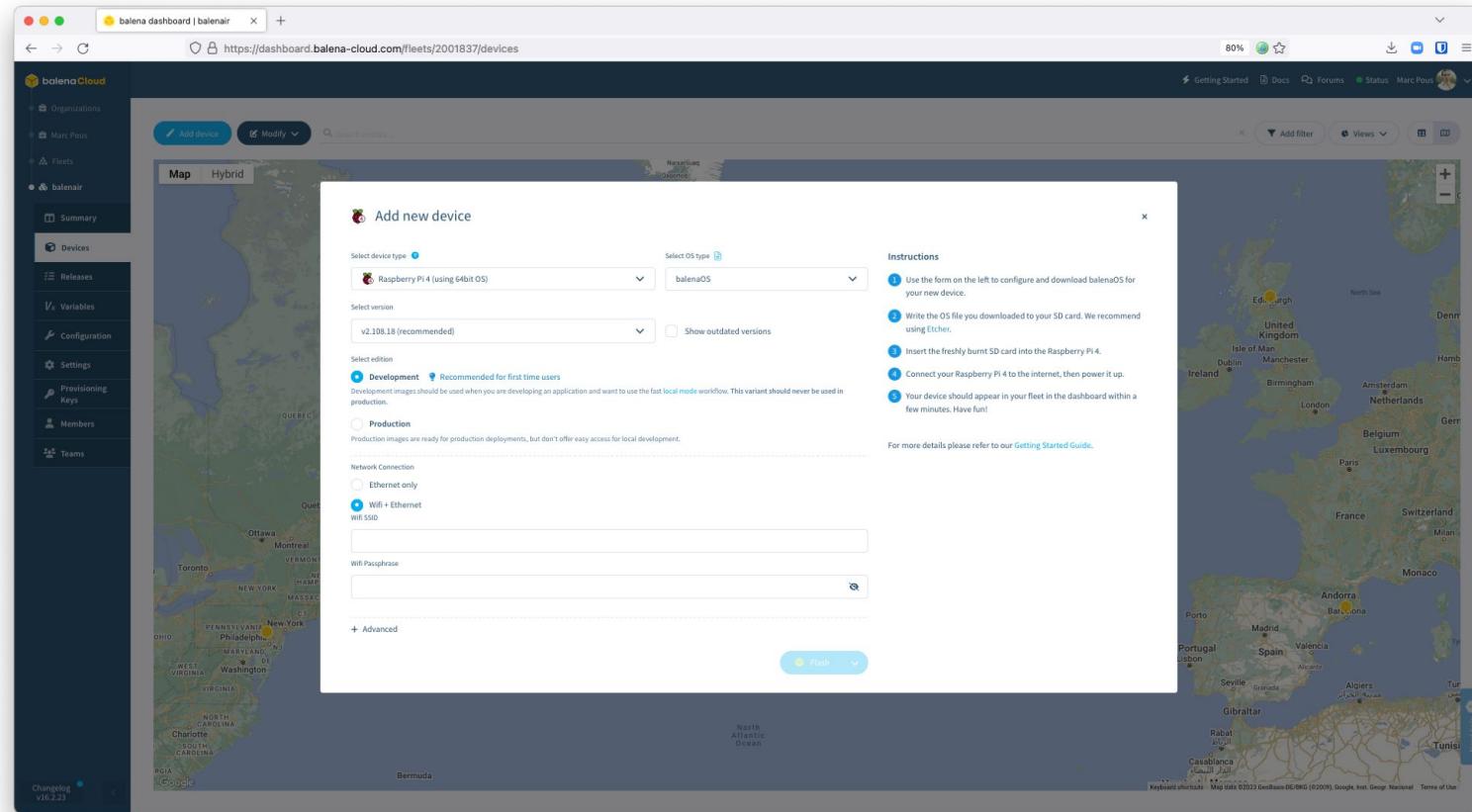
The screenshot shows the balena dashboard interface. On the left, a sidebar menu includes 'Organizations', 'Marc Pous', 'Fleets' (selected), 'Summary', 'Devices', 'Releases', 'Variables', 'Configuration', 'Settings', 'Provisioning Keys', 'Members', and 'Teams'. A 'Changelog' section at the bottom indicates 'v16.2.23'. The main content area displays a fleet named 'balenair'. The fleet summary shows an architecture of 'arch64', a slug of 'narc6/balenair', and a creation date of 'Dec 14th 2022, 12:23 PM'. It also lists '3 Devices' (all online) and '2 Releases' (one tracked). Below this, a map shows the locations of devices across North America and Europe. The map interface includes a legend for device status (green for online, orange for config, blue for updating, red for offline, purple for post-prov, grey for inactive), search bars for 'Add device' and 'Modify', and a 'Search entries...' field.

Deployment with balena

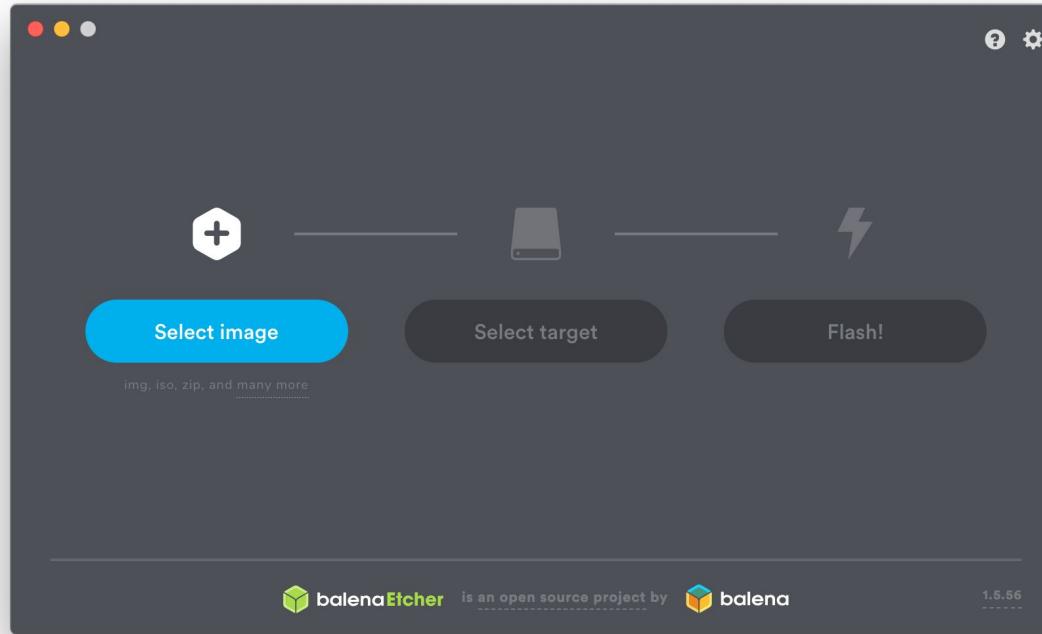
Name	Value	Service	Device overrides	Actions
INFLUXDB_CLOUD_BUCKET_ID	3929d345b1c4376a	All services	1	
INFLUXDB_CLOUD_HOST	https://us-east-1-1.aws.cloud2.i...	All services	1	
INFLUXDB_CLOUD_ORG_ID	28d1f2f565460a6c	All services	1	
INFLUXDB_CLOUD_TOKEN	k-qxb6hTq0xLmzhlm1lVNpZTa...	All services	1	
INFLUXDB_DB	edge	All services		
INFLUXDB_ORG	influxdb	All services		
INFLUX_TOKEN	edge	All services		
LOCATION	default	All services	3	
REPLICATION	<DELETE IF NOT USING REPLIC...	All services		

1 - 9 of 9

Deployment with balena



Deployment with balena



Deployment with balena

The screenshot shows the balena dashboard interface for a device named "UK".

Device Summary:

- Status: Online
- UUID: 22d3415
- Type: Raspberry Pi 4 (using 64bit OS)
- Online for: 1 day
- Host OS Version: balenaOS 2.107.10+rev3
- Supervisor Version: 14.4.4
- Fleet: marc/balenaIR
- Local IP Address: 192.168.1.223
- Public IP Address: 77.181.217.48
- MAC Address: E4:5F:81:ED:42:E7
- Tags: None
- Notes: Add device notes

Services:

Service	Status	Release
connector	Running	dcf6781
dashboard	Running	dcf6781
lag	Running	dcf6781
influxdb2	Running	dcf6781
mqtt	Running	dcf6781
wifi-connect	Running	dcf6781

Logs:

```
12.01.23 15:55:01 (+0000) dashboard Waiting for Influx API
12.01.23 15:55:06 (+0000) dashboard Waiting for Influx API
12.01.23 15:55:11 (+0000) dashboard Waiting for Influx API
12.01.23 15:55:16 (+0000) dashboard Waiting for Influx API
12.01.23 15:55:21 (+0000) dashboard Waiting for Influx API
12.01.23 15:55:27 (+0000) dashboard Waiting for Influx API
12.01.23 15:55:32 (+0000) dashboard Waiting for Influx API
12.01.23 15:55:37 (+0000) dashboard Waiting for Influx API
12.01.23 15:55:42 (+0000) dashboard Waiting for Influx API
12.01.23 15:55:46 (+0000) wifi-connect Checking Internet connectivity ...
12.01.23 15:55:46 (+0000) wifi-connect Your device is already connected to the Internet.
12.01.23 15:55:46 (+0000) wifi-connect Skipping setting up WiFi-Connect Access Point. Will check again in 120 seconds
12.01.23 15:55:47 (+0000) dashboard Waiting for Influx API
12.01.23 15:55:52 (+0000) dashboard Waiting for Influx API
```

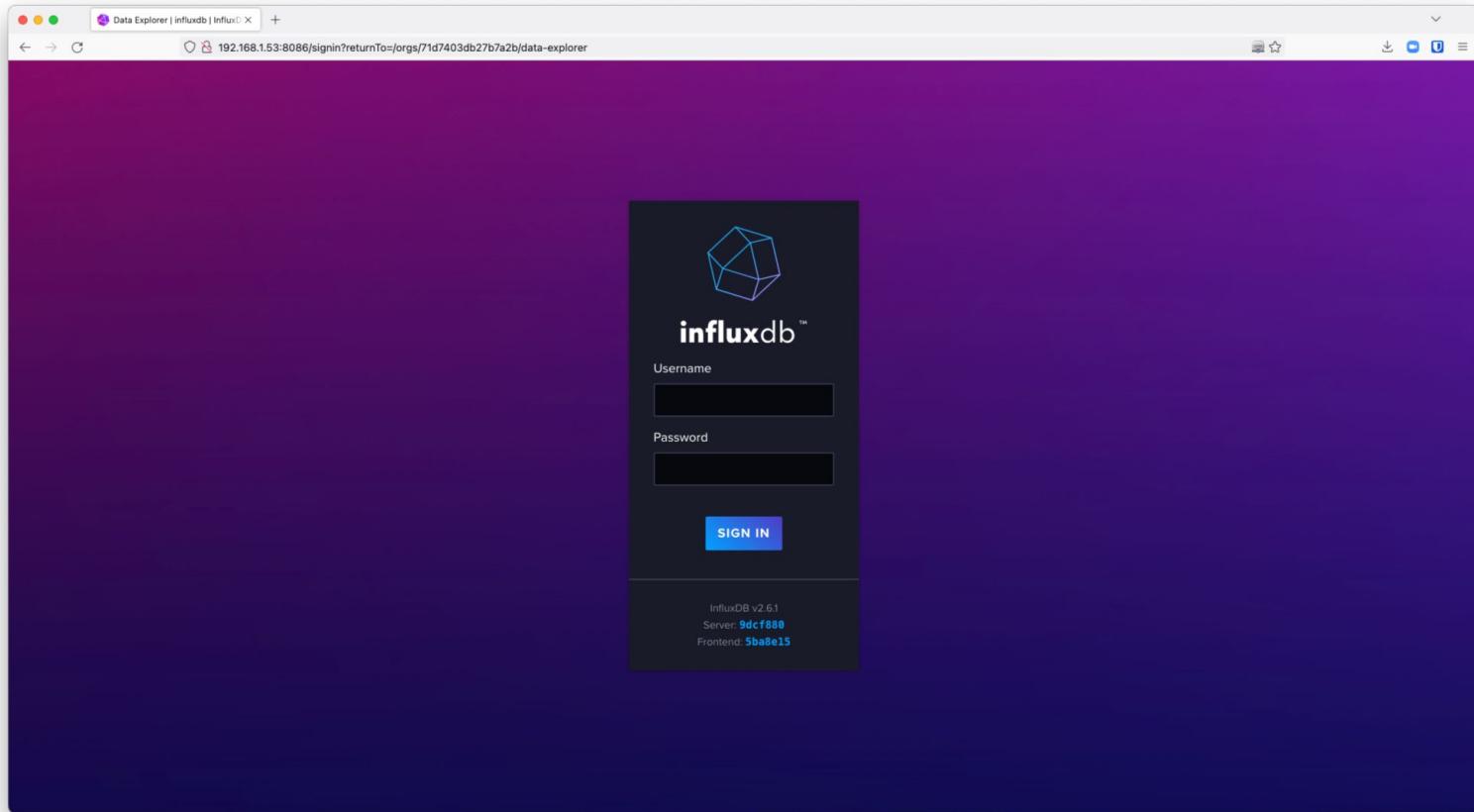
Terminal:

Select a target:

Deployment with balena



Deployment with balena



InfluxDB Cloud



InfluxDB Cloud

```
import
"contrib/anaisdg/anomalydetection"

from(bucket: "C02-Demo")
    |> range(start: v.timeRangeStart,
stop: v.timeRangeStop)
    |> filter(fn: (r) => r["_measurement"] ==
"mqtt_consumer")
    |> filter(fn: (r) => r["_field"] == "co2")
    |> anomalydetection.mad(threshold: 3.0)
    |> filter(fn: (r) => r["level"] == "anomaly")
// Notice how you can filter for the actual MAD value to better discover anomalous series.
    |> filter(fn: (r) => r["MAD"] > 3.0)
    |> group(columns: ["location"],
mode:"by")
    |> aggregateWindow(every: 1m, fn:
count, createEmpty: false )
    |> yield()
```

```
from(bucket: "C02-Demo")
    |> range(start: v.timeRangeStart,
stop: v.timeRangeStop)
    |> filter(fn: (r) => r["_measurement"] ==
"mqtt_consumer")
    |> filter(fn: (r) => r["_field"] == "co2")
    |> filter(fn: (r) => r["location"] == "ES")
    |> aggregateWindow(every: 30s, fn:
mean, createEmpty: false )
    |> yield(name: "actual")
    |> holtWinters(n: 8, interval: 30s,
withFit: true)
    |> yield(name: "forecast")
```