

EdgeX Foundry(TM) - Build Simple Data Monitor UI #01

<https://www.topcoder.com/challenges/4c8811c4-6504-40a1-a9d0-ad25ee7c1af7>

Deployment and Verification Document

Prerequisites

- Go 1.11+ (go modules)
- Docker

Setup

You'll need some Fyne tools, specifically fyne-cross for cross-platform building. In order to install them just run

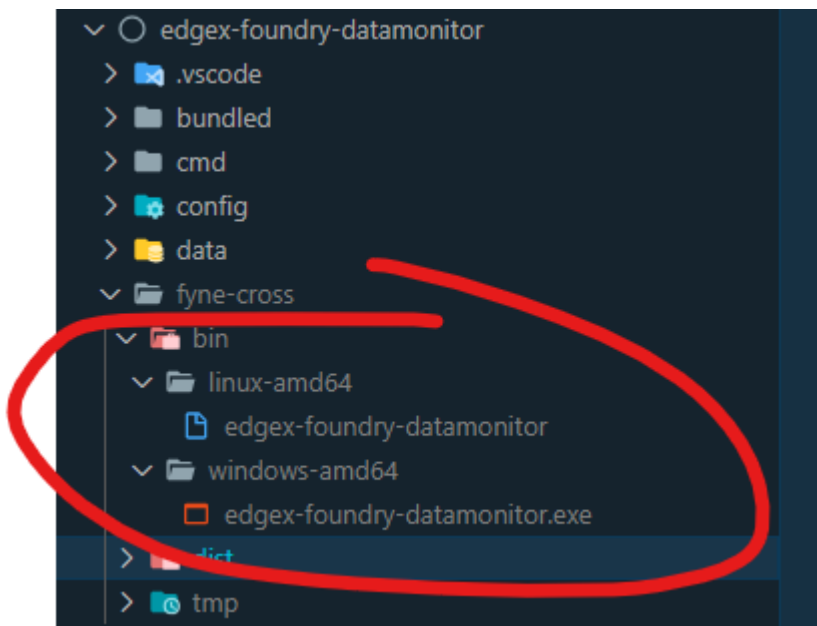
make install-deps

The included README.MD provides additional info

Verification

Run "make cross-compile" inside the root folder and the application will be build for windows and linux, please read the notes about Darwin build/deployment

The executables will be available here:



Run “docker compose up” (or “docker-compose up” depending on how updated is your installation).

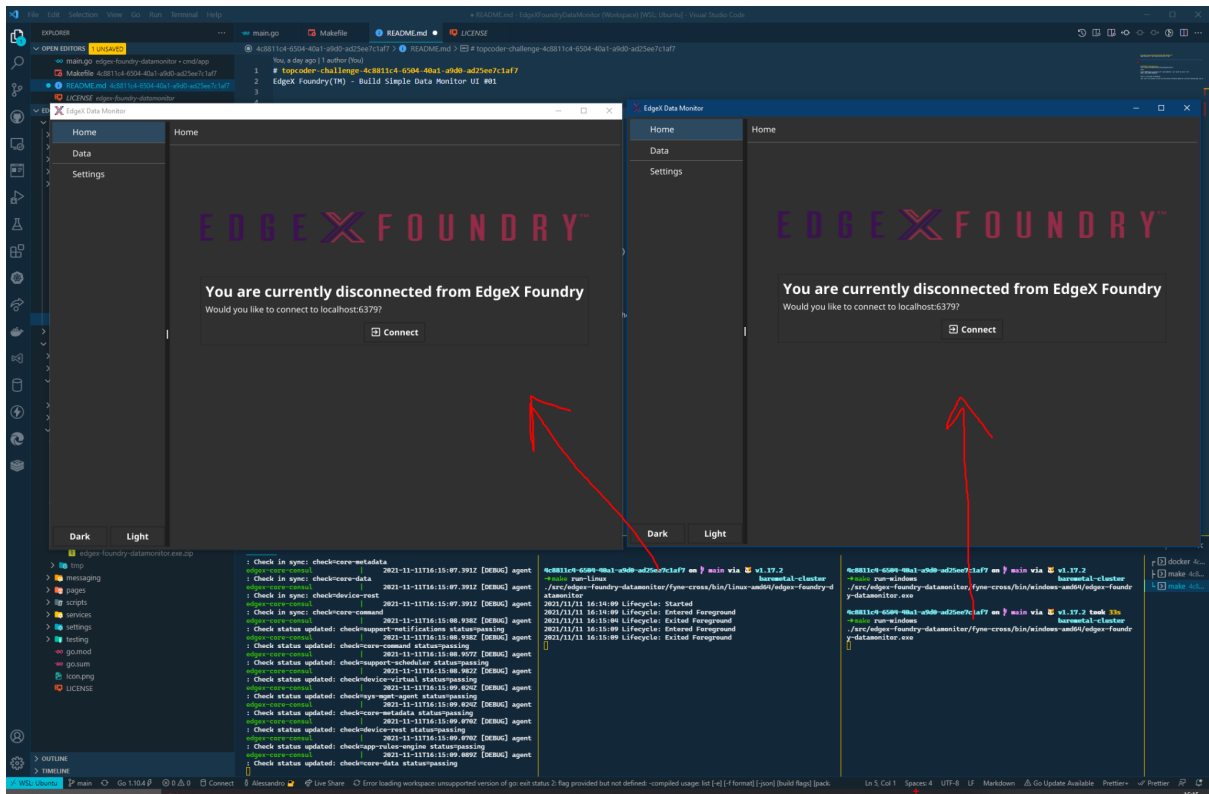
Then, depending on your platform, you should be able to run a linux or a windows executable.

I have added shortcuts in the makefile so you can do:

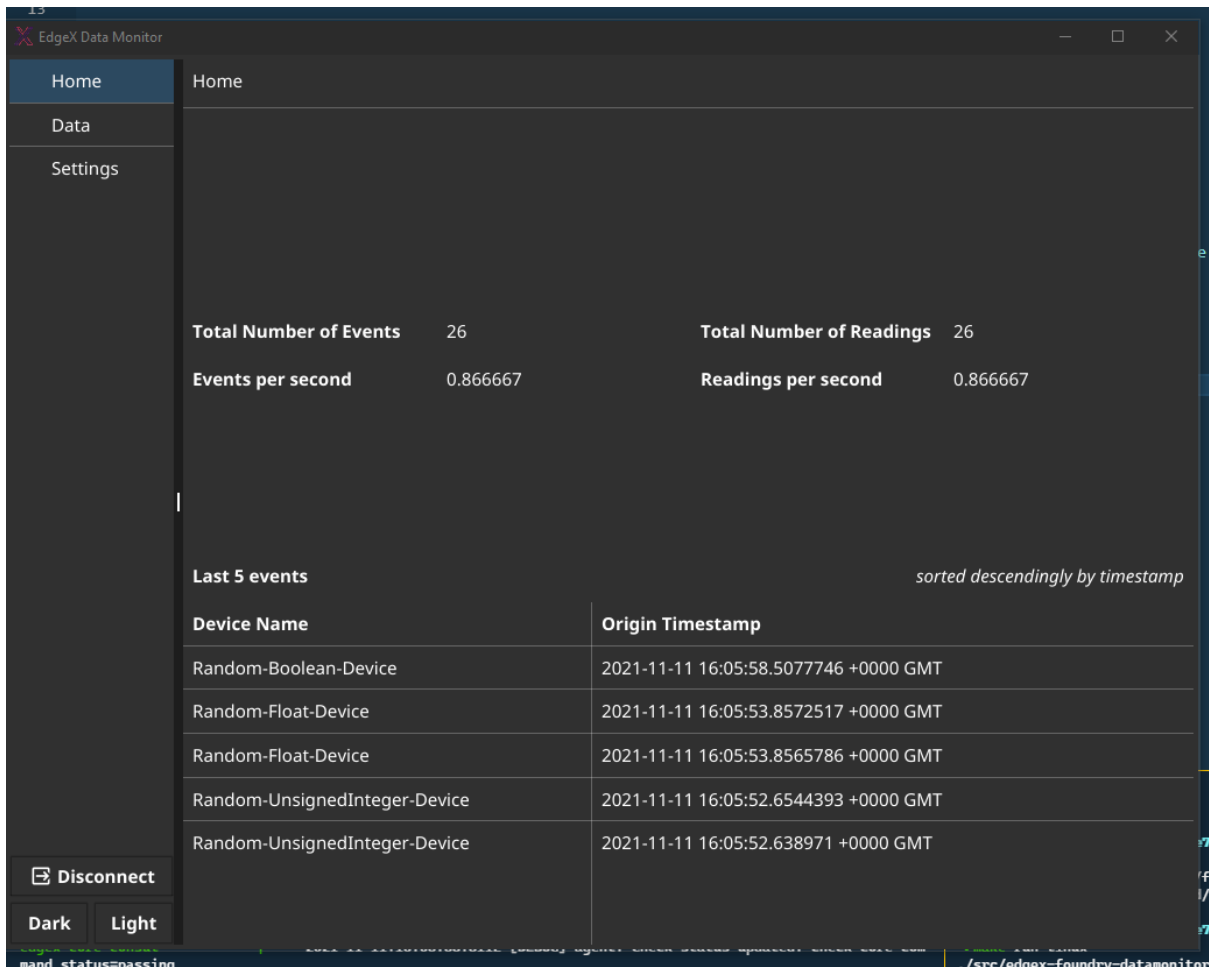
“make run-linux”

or

“make run-windows”



After clicking on connect you should see something like this:



In order to verify that the events are coming through and that the calculations are done correctly you can navigate to

<http://localhost:4000/#/metadata/device-center/device-list>

Click on edit

The screenshot shows a web application interface with a dark sidebar on the left containing navigation links: Dashboard, System, Metadata (highlighted), DataCenter, Scheduler, Notifications, RuleEngine, and AppService. The main content area is titled 'Device List' and shows a breadcrumb 'Metadata > Device Center > Device List'. Below the title are three tabs: 'Device Service', 'Device' (selected), and 'Device Profile'. A toolbar contains 'Refresh', '+ Add', 'Edit', and 'Delete' buttons. A table lists devices with columns for ID, Name, and Description. The first device is selected with a checkbox.

ID	Name	Description
<input checked="" type="checkbox"/> fa730472-23ee-4f93-94f8-43bca5b44091	Random-Float-Device	Example of Device Virtu
<input type="checkbox"/> d9d18045-ebbf-445c-85ed-4771c46f7259	sample-json	RESTful Device that sen
<input type="checkbox"/> bf225c5d-9e7b-470e-a084-0521368a49e1	sample-image	RESTful Device that sen
<input type="checkbox"/> ad3a1492-c30a-4bd5-bd08-e7b7b346c818	sample-numeric	RESTful Device that sen

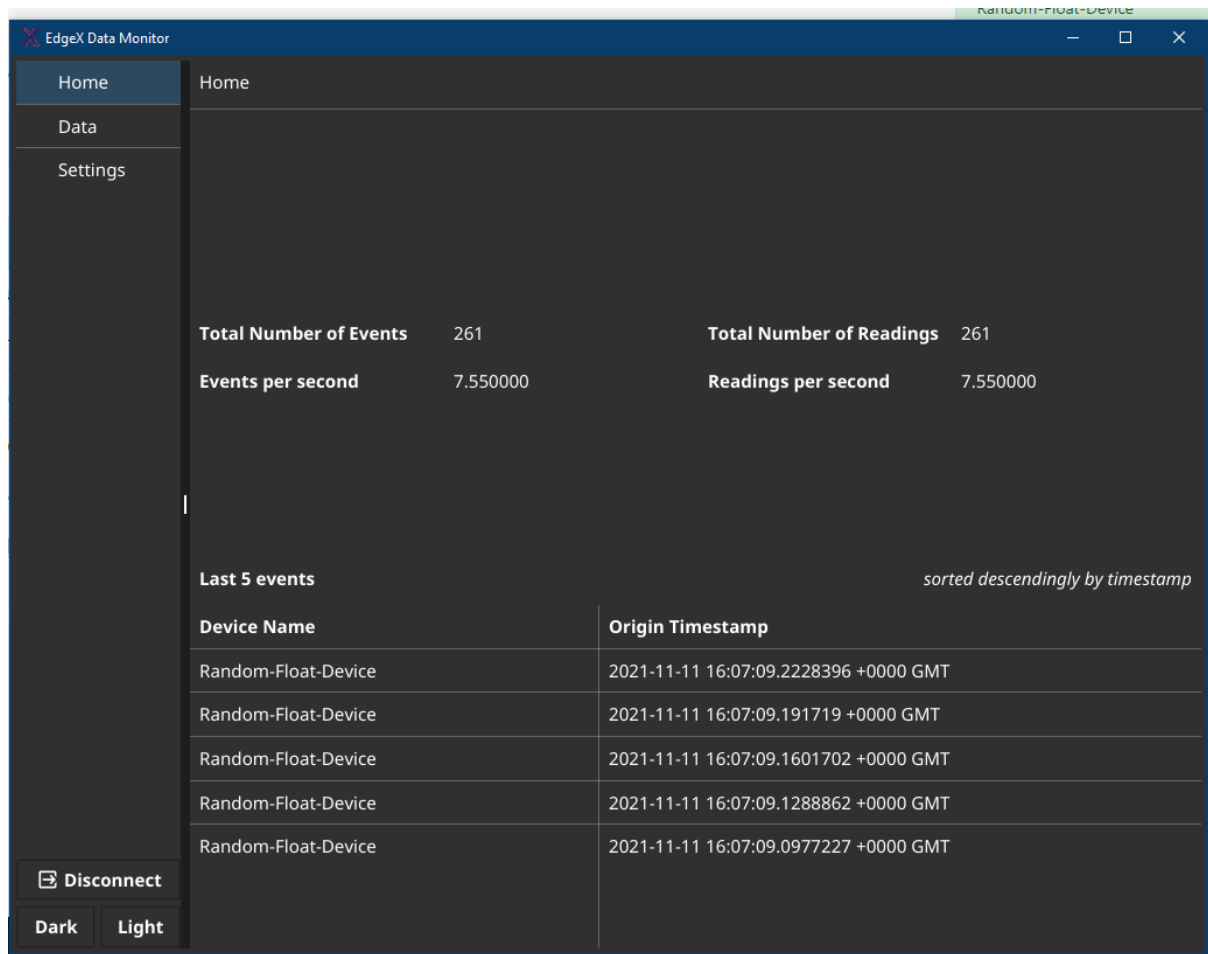
Scroll below in the AutoEvents section

The screenshot shows the 'AutoEvents' configuration section. It has a header with a red icon and the text 'AutoEvents'. Below the header is a button 'Add More AutoEvent'. The configuration area contains three rows of settings. The first row has 'Interval' set to '30' with a green checkmark and 'second' with a dropdown arrow. The second row has 'onchange' set to 'false' with a dropdown arrow. The third row has 'resource' set to 'Float32' with a green checkmark and a dropdown arrow. A red 'Remove' button is located below the third row. The second row of the configuration area is partially visible at the bottom of the screenshot.

Interval	onchange	resource
30 ✓ second	false	Float32 ✓
30 ✓ second		

And adjust one of the values, for example setting to 1 second.

You'll notice that the counters will show an increase and the last events will show more instances of the Device you changed.



I tried setting crazy values like 30 milliseconds and even if there are a few graphical glitches here and there, It works both on windows and linux (WSL2 - Ubuntu)

Notes

- I had to patch the referenced <https://github.com/edgexfoundry/go-mod-messaging> library because it uses a library that made me lose a whole day while trying to make it work in my environment. It will be soon deprecated as stated here <https://github.com/edgexfoundry/go-mod-messaging/issues/73>

Also, we use Redis, not zeromq so it's totally fine

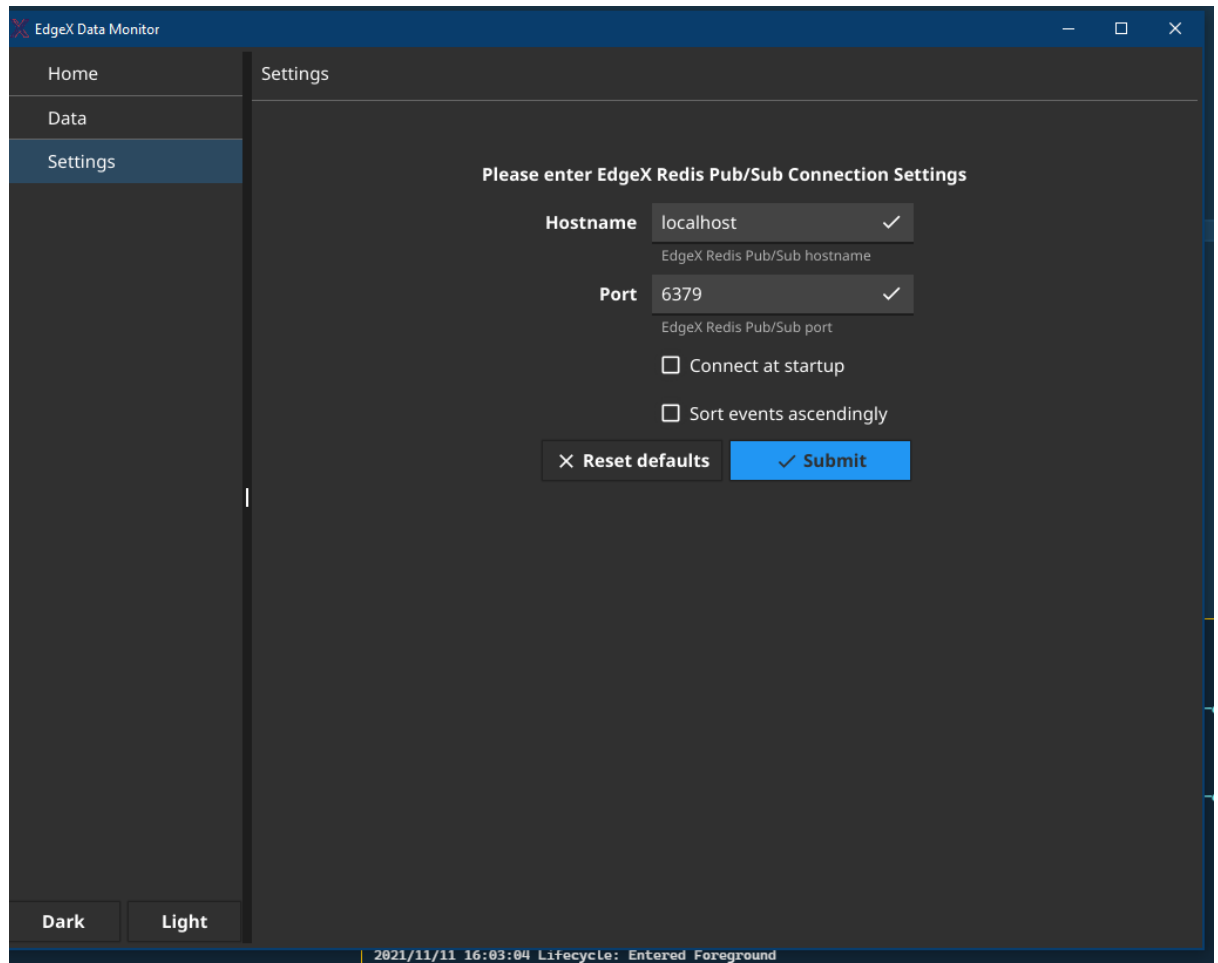
Currently my patched version is referenced with a go mod replace entry

```
replace github.com/edgexfoundry/go-mod-messaging/v2 v2.0.1 => github.com/deblasis/go-mod-messaging/v2 v2.0.2-dev.7.0.20211109140946-fc6f30b82981
```

It's a trivial set of changes that I might contribute upstream, however my repository is

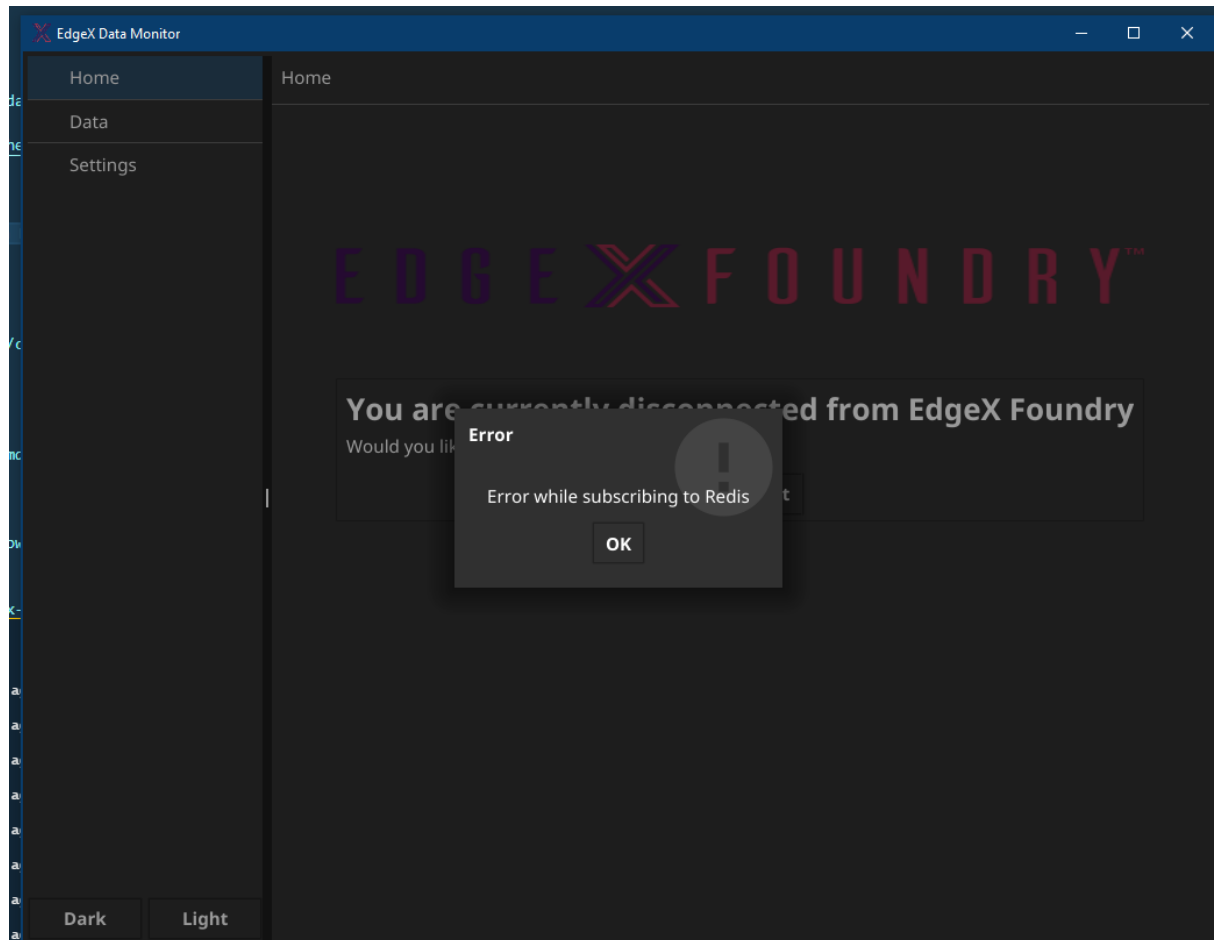
public and also included in this package

- I have added a few features like additional settings



That should be pretty self-explanatory

- Error message if unable to connect



- Configurable sorting of the events
- Disconnect button