DC Motor Integration in BEMOSS

Brian Lauer
Advisor: Dr. Suruz Miah

Department of Electrical and Computer Engineering
Bradley University
1501 W. Bradley Avenue
Peoria, IL, 61625, USA

Friday, August 14, 2019

BRADLEY University

Outline

BRADLEY University

Device API Translator

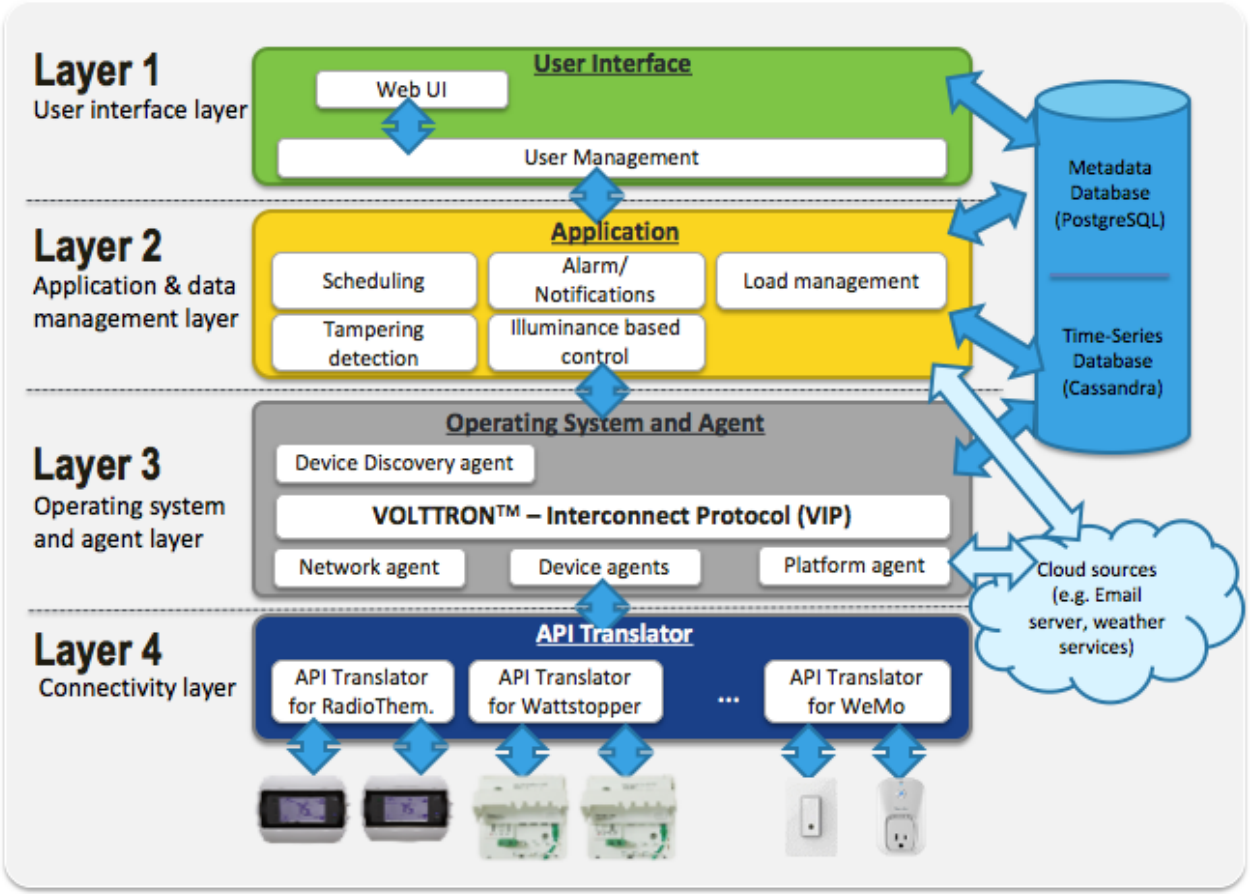

Figure: Courtesy of BEMOSS developer resources

Steps in adding device API

Methods to implement

- `API_info` returns list of supported device attributes (`device_model`, `vendor_name`, `html_template`, `chart_template` etc.)
- `dashboard_view` returns parameters of device to be shown on the BEMOSS dashboard
- `ontology` maps parameters from device to BEMOSS parameters
- `discover` is called by the device discovery agent and returns attributes of discovered devices
- `getDataFromDevice` returns dictionary of device data
- `identifyDevice` visually identifies the device, triggered by web UI and sends message to control agent
- `setDeviceStatus` sends control commands to device

BRADLEY University

Steps in adding device API

Methods in motor API

- `argsToPi` uses ssh to run `XBEEcontrol.py` with arguments to control motor and retrieve data
- `toggleDeviceStatus` queries the device for on/off status then changes settings of the motor, uses `getDataFromDevice` and `setDeviceStatus`

Steps in adding new device type

Steps

- Select a device type id (1, 2, 3, 4 and 7 are reserved)
- Register new device type in BEMOSS db by updating `PROJECT_DIR/Web_Server/defaultDB.py` then running the script
- Update `discover_devices` function in `Web_Server/webapps/discovery/views.py`
- Add widget for new device in `Web_Server/webapps/discovery/templates/discovery/manual_discovery.html`
- Update `Web_Server/static/app_js/discovery.js` with code to enable use of the "Set all to 'Approved'" button in the Device Discovery page

BRADLEY University

Steps in adding new device type

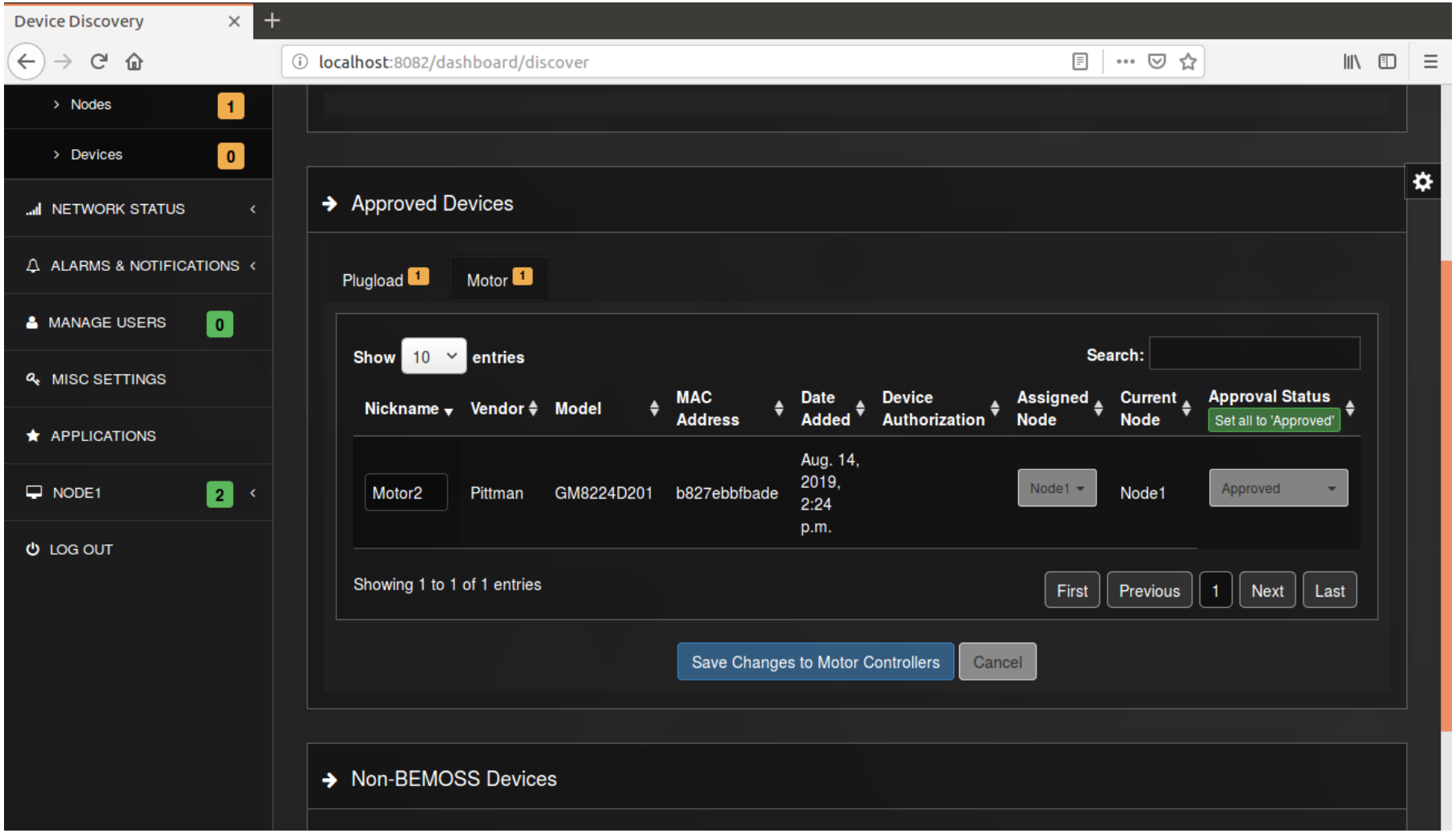

Figure: Device discovery page

Problems Fixed

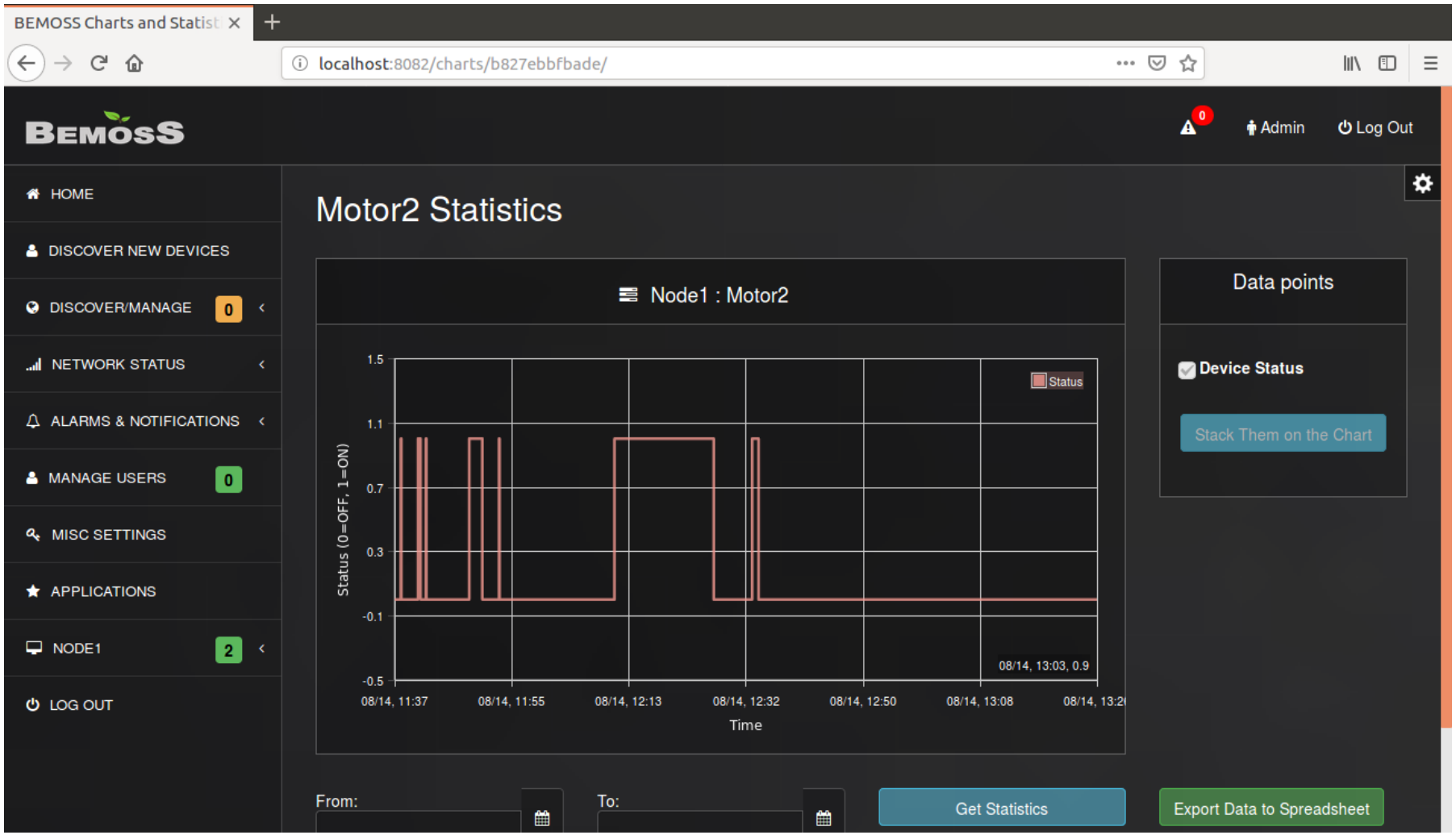

Figure: Fully functional jQuery plot

Additions

ssh keys

- `ssh-keygen` generates a public and private key
- Public key copied to RPi, private key kept on host machine
- More secure than using password
- `https://www.raspberrypi.org/documentation/remote-access/ssh/passwordless.md`

Priorities

- Speed control algorithm
- Finish paper
- Still looking for new device