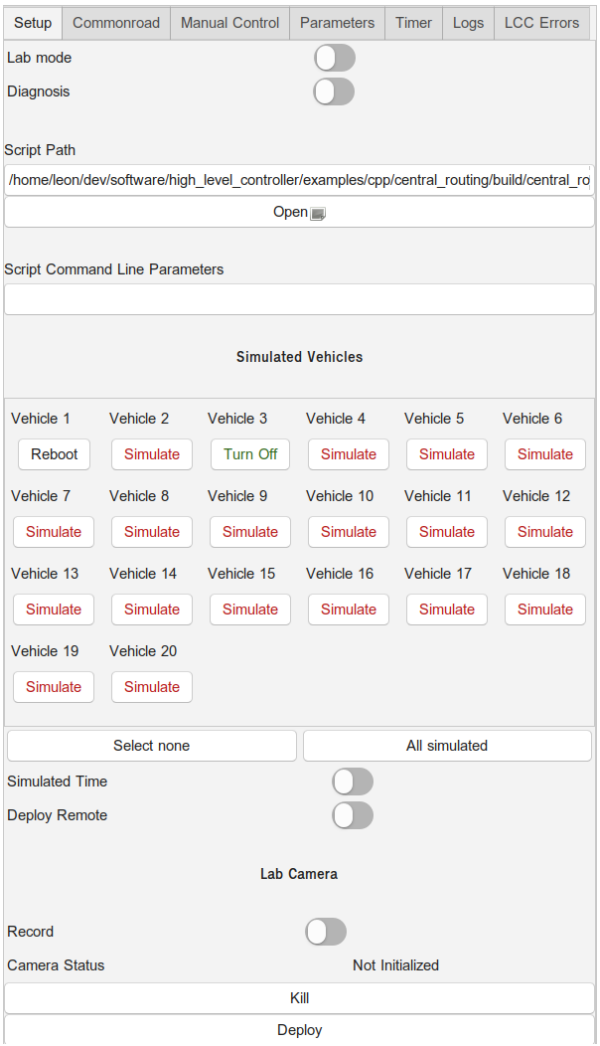
Setup (Tab) (WIP)



The setup view allows to launch all components required to perform a simulation from within the LCC's GUI. It can also be used for real vehicles, and to deploy your selected program or script remotely on the NUCs (HLCs).

> **Deploy Remote** is now called **Deploy Distributed**

Selecting a Script

A single script can be selected by editing the path text field or selecting a file using the *Open* button. In the file chooser, you can select whether you want to use an executable or a Matlab script in a drop down menu at the bottom right of the window.

The file chooser remembers your previously selected file.

If no script is selected (empty string as script path), the LCC will only start the Middleware. You can run your script yourself outside of the LCC, connect to the Middleware and then start the simulation.

Required Script Properties

Your script should take the following arguments:

C++	Matlab

`--vehicle_ids=`
`--dds_domain=`
`--middleware_domain=` **`(nur im eProsima Branch)`**
`--node_id=`
`--simulated_time=`
`--dds_initial_peer=`
	RTI DDS: (your_params, varargin)
	eProsima: (your_params, middleware_domain_id, varargin)

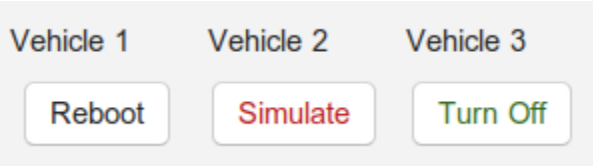
Most of the required C++ parameters are used to set values for the cpm library. Of these parameters, you are probably only interested in the list of vehicle IDs your program is responsible for (--vehicle-ids). The same is true for the matlab script, where varargin gives you a list of vehicle IDs - any other parameters, which come beforehand, are custom.

The Middleware domain ID can currently only be set consistently in the **eProsima** Branch, the functionality might not be added to support **RTI** w.r.t. Matlab, although the change would be rather small (you can adapt your Matlab scripts accordingly), as the Lab will change to using eProsima only in the future.

Further parameters can be specified in the text field below the *Open* button. If you need to access these command line parameters in your C++ program, simply use the Command Line Parser and --your_param= in the text field. Custom arguments can be set by yourself this way during testing / for your simulations. Matlab custom arguments just need to be passed in a comma-separated fashion (e.g.: 1, 155, 3).

Selecting Vehicles

You can select, for each vehicle individually, whether they should be simulated, or control their current state if they are real vehicles.



Simulated Vehicle	• *Simulate* The vehicle is currently not simulated, but can be started with a button press • *Turn Off* The vehicle is currently simulated and can be turned off by pushing the button
Real Vehicle	• *Reboot* The vehicle was detected as real vehicle. By using this button, you can send a reboot command to it to reboot the Raspberry Pi.

The buttons below the scrollable view allow you to set the state for all (non-real) vehicles at once.

Vehicles that are set to be off will be ignored by your script, as their IDs are not passed to it. You do not need to start the simulated vehicles - the LCC does that for you immediately after selection. They are killed when the LCC is closed or when another mode (real or off) is chosen.

> **Warning**
>
> If you have not started the LCC with the *correct parameters (e.g. use the main server DDS domain)*, you might not be able to connect to / see the real vehicles.

Simulated Time

This switch should be self-explanatory - if you want to use simulated time, turn it on, else turn it off.

Lab mode

The IPS tracks the real vehicles, so it should be turned on if real vehicles are used. In any other case, it should stay turned off.

Deploy Distributed

This option includes a lot of features, thus we created an own page for it. If this switch is set, your program gets uploaded to the NUCs / HLCs for deployment, instead of being deployed locally on your machine. One script instance handles one vehicle (ID) per NUC / HLC.

Lab Camera

Turn the "Record" switch on before clicking "Deploy" to record your experiment.

After stopping the experiment with a click on "Kill", you will find the video file in `/tmp/cpm_lab_recordings` .

If you are planning to play the video on windows without installing codecs, convert the video e.g. with

```
ffmpeg -i $INPUT_NAME_WITH_FILE_EXTENSION$ -ss $START_IN_SECONDS$ -t $DURATION_IN_SECONDS$ -c:v libx264 -strict
-2 -preset slow -pix_fmt yuv420p -f mp4 output.mp4
```

Deploy / Kill

When you have finished your setup, you can launch all the components / scripts by clicking on *Deploy*. Not all software starts immediately then. Some, e.g. those that rely on the Middleware, wait until you start the simulation in the Timer Tab. (Wait for the program to show up there before you start the timer). Due to initialization and synchronization, it makes more sense to wait for all participants before starting the timer.

Local: This starts several tmux sessions that run the required scripts. These sessions are terminated when you press on *Kill*. A stop signal is sent to all timers when the kill button gets pressed.

Remote: The folder in which the script is located gets uploaded to the NUCs if you choose to deploy the software remotely. Each HLC/NUC script gets assigned a single vehicle ID. IDs are given in ascending order regarding the selected vehicle IDs and the available HLC IDs. Not all vehicles might be controlled, if not enough NUCs are online. Middleware and Script are deployed on the HLCs, simulated vehicles, IPS and CloudDiscoveryService are deployed on the local machine.

Important

To execute your software, you often need to Deploy Switch to Timer Tab Wait for the Middleware to show up Start the Timer

Important

Always use Kill after you have finished, or else some software might end up running on a system (like the NUC / HLC) until it is restarted.