Lab Software Preparation

In case you are unsure that the most recent build process performed on your PC built your desired version of the CPM Lab software, follow these steps.

Open a terminal, navigate to the software repository and checkout your desired branch or tag. If unsure, checkout the master branch. If there are updates available, download them, clean the repo and build the software again. Substitute `<BRANCH_OR_TAG>` in the following code!

```
cd ~/dev/software
git fetch
git checkout <BRANCH_OR_TAG>
rm -rf cpm_lib/dds_idl_cpp
rm -rf cpm_lib/dds_idl_matlab
```

If you are working on the Main PC of the lab, build the software without the `--simulation` switch. Otherwise, build all the necessary files to simulate the CPM Lab:

```
bash build_all.bash --simulation
```

What to Consider After an Update

Some common problems arise after an update. The .idl files for the communication might have changed, and the files for the NUCs or vehicles might be different as well. Thus, if you pull a new version (of the master), make sure that you do the following:

- **Delete** the auto-generated files created from the .idl files. You can find them in `cpm_lib/dds_idl_cpp` and `cpm_lib/dds_idl_matlab`. We do not provide a mechanism that looks for updates in the .idl files, and thus, if they have been updated, do not re-generate them if old versions are still present. If they do not get updated, the build or communication might fail.
- **Re-build** the cpm_lib, Lab Control Center, Mid Level Controller, Middleware etc. The build_all script should cover all these steps.
- For the NUCs: Make sure that the NUC packages were actually updated properly. You can find out more here in *Provide all required packages*.