DART RACING

Streamlining Development: Marrying the Code Base and GUI Simulation Software to Docker

Short Talk by Oscar Fritzsche

DART RACING

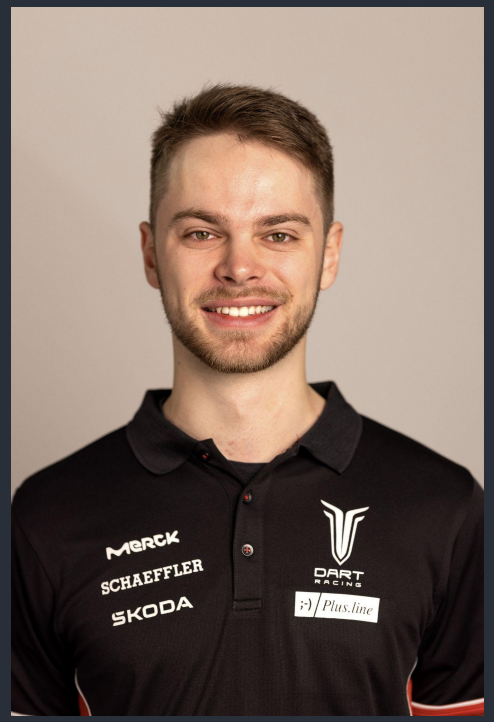
Streamlining Development: Marrying the Code Base and GUI Simulation Software to Docker

Or: Why Good Developer Documentation is Crucial

Short Talk by Oscar Fritzsche

About Me

- Oscar Fritzsche
- B.Sc. Computer Science @ TU Darmstadt
- Member of the Autonomous System Department at DART Racing
- Responsible for:
  - Trajectory Optimization
  - Development Setup
  - PC Hardware



Agenda

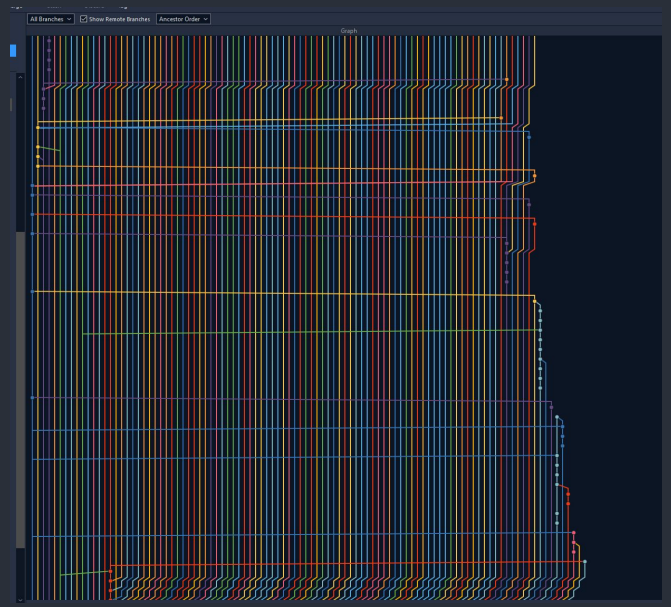
- Motivation
- Docker + Dev Containers
- Where the Magic Happens
- Short Demo
- Advantages & Drawbacks

Motivation

Shitty installation instructions, anyone?

Motivation

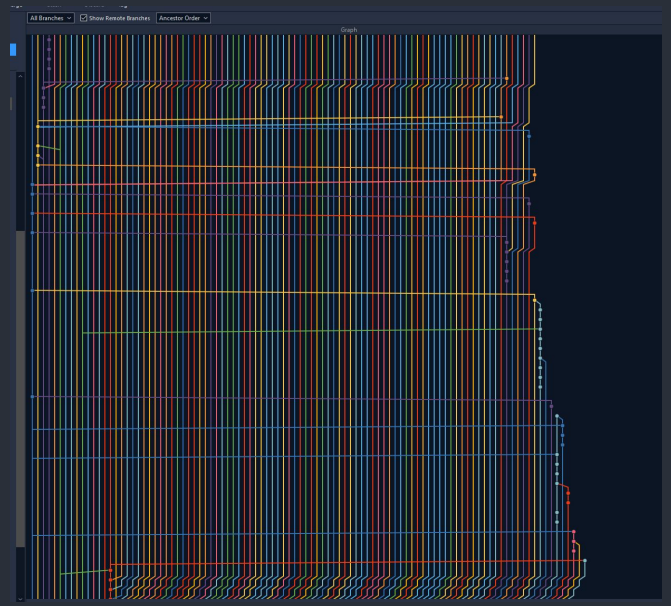
Shitty installation instructions, anyone?



https://www.reddit.com/r/ProgrammerHumor/comments/12f7ye2/i_see_a_lot_of_screenshots_of_horribly_complex/#lightbox

Motivation

Shitty installation instructions, anyone?



https://www.reddit.com/r/ProgrammerHumor/comments/12f7ye2/i_see_a_lot_of_screenshots_of_horribly_complex/#lightbox

Prerequisites

- You have to install ROS Noetic for Ubuntu 20.04 LTS and source it.
- (You need to install git-lfs.)
- To make it as convinient as possible add a SSH-Key to your Bitbucket account.
- If you are using conda, make sure to deactivate it while running ROS: `conda deactivate`

Addidtional needed packages

```
sudo apt install python3-catkin-tools
```

If you need to update CMake

```
wget -O - https://apt.kitware.com/keys/kitware-archive-latest.asc 2>/dev/null | sudo apt-key add -
sudo apt-add-repository 'deb https://apt.kitware.com/ubuntu/ bionic main'
sudo apt-get update
sudo apt install cmake
```

Motivation

- Shitty installation instructions
- Accidentally broke the installation *(again…)*
- Knowledge loss because someone left the team

→ Move fast, get work done, get the car to move even faster

Enter: Docker + Dev Containers

https://www.docker.com/company/newsroom/media-resources/
https://downloads.marketplace.jetbrains.com/files/21962/701151/icon/default.png

Enter: Docker + Dev Containers

How do you deal with GUI software inside a Container?

https://www.docker.com/company/newsroom/media-resources/
https://downloads.marketplace.jetbrains.com/files/21962/701151/icon/default.png

Dev Containers

Built on top of a Docker image

```
# Switch from root to user
USER $USERNAME
# ros stuff
RUN git lfs install && \
    sudo mkdir -p /etc/apt/keyrings && \
    curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc > /tmp/ros.asc && \
    sudo -H gpg -o /etc/apt/keyrings/ros.gpg --dearmor /tmp/ros.asc && \
    sudo sh -c 'echo "deb [signed-by=/etc/apt/keyrings/ros.gpg] http://packages.ros.org/ros/ubuntu fo
    sudo apt-get update -y && \
    sudo apt-get install -y --no-install-recommends ros-noetic-desktop-full ros-noetic-rosmon && \
    echo "source /opt/ros/noetic/setup.bash" >> ~/.bashrc && \
    source ~/.bashrc && \
    sudo apt-get install -y --no-install-recommends python3-rosdep python3-rosinstall python3-rosinst
    sudo rosdep init && \
    rosdep update && \
    sudo apt-get install -y --no-install-recommends python3-catkin-tools && \
# cmake
    wget -O- https://apt.kitware.com/keys/kitware-archive-latest.asc > /tmp/kitware.asc && sudo -H gp
    sudo sh -c "echo 'deb [signed-by=/etc/apt/keyrings/kitware.gpg] https://apt.kitware.com/ubuntu/ f
    sudo apt-get update -y && \
    sudo apt-get install -y --no-install-recommends cmake && \
# Libtorch
    curl https://download.pytorch.org/libtorch/cpu/libtorch-cxx11-abi-shared-with-deps-2.0.1%2Bcpu.zi
    unzip /tmp/libtorch.zip -d ~/libtorch && \
    echo "export Torch_DIR=$HOME/libtorch/libtorch" >> ~/.bashrc && \
    source ~/.bashrc && \
# Trajectory Optimizer Dependencies
    echo 'export PATH="$PATH:$HOME/.local/bin"' >> ~/.bashrc && source ~/.bashrc && \
    sudo apt-get install -y --no-install-recommends python3-pip && \
    python3 -m pip install --upgrade pip && \
    python3 -m pip install --upgrade numpy matplotlib \
    trajectory_planning_helpers casadi scipy scikit-learn rospkg jax jaxlib && \
# clean up
    sudo apt-get -y clean && \
    sudo rm -rf /var/lib/apt/lists/* && \
    rm -rf /tmp/*


# image optimization
FROM scratch
```

Further configuration via devcontainer.json

```
{
    "name": "DART-AS-Devcontainer-Base",
    "image": "dart-dev-base",
    "runArgs": [
        "--privileged",
        "--network=host"
    ],
    "containerUser": "dart-dev",
    "workspaceMount": "source=${localWorkspaceFolder},target=/home/dart-dev/${localWorkspaceFolderBas
    "workspaceFolder": "/home/dart-dev/${localWorkspaceFolderBasename}",
    "mounts": [
        "source=${localEnv:HOME}${localEnv:USERPROFILE}/.bash_history,target=/home/dart-dev/.bash_his
        "source=profile,target=/home/dart-dev,type=volume"
    ],
    "features": {
        "ghcr.io/devcontainers/features/desktop-lite:1": {
            "version": "latest",
            "noVncVersion": "1.2.0",
            "password": "noPassword",
            "webPort": "6080",
            "vncPort": "5901"
        }
    },
    "forwardPorts": [6080, 5901],
    "portsAttributes": {
        "6080": {
            "label": "desktop-browser"
        },
        "5901": {
            "label": "desktop-VCN"
        }
    },
    "onCreateCommand":
        "cd ~/${localWorkspaceFolderBasename}; sudo apt-get update -y && rosdep update && rosdep inst
    "postCreateCommand":
        "git config --global --add safe.directory ${containerWorkspaceFolder} && echo 'source /home/c
}
```

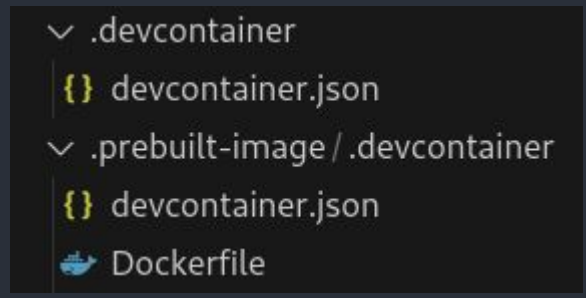
Dev Containers

- Prebuilt image
  → Faster startup for everyone else
- Upload image to Docker Hub

Dev Containers

- Prebuilt image

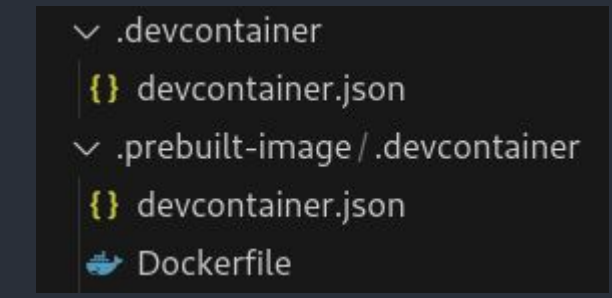
  → Faster startup for everyone else
- Upload image to Docker Hub
  - Don’t put an 8 GB image into git.



Dev Containers

- Prebuilt image

  → Faster startup for everyone else
- Upload image to Docker Hub
  - Don’t put an 8 GB image into git.
  - Don’t be stupid.



Dev Containers

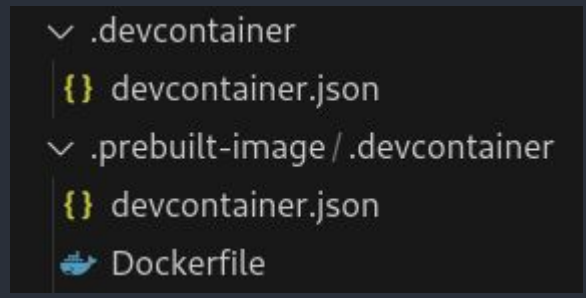
- Prebuilt image

  → Faster startup for everyone else
- Upload image to Docker Hub
  - Don’t put an 8 GB image into git.
  - Don’t be stupid.
  - Don’t be like me.

Dev Containers

- Prebuilt image

  → Faster startup for everyone else
- Upload image to Docker Hub
  - Don't put an 8 GB image into git.
  - Don't be stupid.
  - Don't be like me.



- Completely unrelated: https://rtyley.github.io/bfg-repo-cleaner/
  - Great tool for getting rid of large files in your commit history!

DART RACING

Where Does the Magic Happen?

Where the Magic Happens

```
"features": {
    "ghcr.io/devcontainers/features/desktop-lite:1": {
        "version": "latest",
        "noVncVersion": "1.2.0",
        "password": "noPassword",
        "webPort": "6080",
        "vncPort": "5901"
    }
},
"forwardPorts": [6080, 5901],
"portsAttributes": {
    "6080": {
        "label": "desktop-browser"
    },
    "5901": {
        "label": "desktop-VCN"
    }
},
```

DART RACING

Demo Time

Advantages & Drawbacks

Drawbacks:

- Creating the base image can be *really* annoying
- The provided desktop environment is extremely bare-bones
- Images are huge (maybe a skill issue on my part)
- Running the container is resource intensive
  - Our setup requires at least 12 GB of RAM
- Windows containers not supported

Advantages & Drawbacks

Advantages:

- Easy onboarding once initial setup is done
- Fast recovery of broken installations
- No dependency hell (“But it works on my machine!”)
- There is always a reproducible working state
- Works with any GUI software you might need
- Windows containers not supported

Drawbacks:

- Creating the base image can be *really* annoying
- The provided desktop environment is extremely bare-bones
- Images are huge (maybe a skill issue on my part)
- Running the container is resource intensive
  - Our setup requires at least 12 GB of RAM
- Windows containers not supported

Thank you very much for listening!

Feedback is much appreciated: