Machinekit Projects Overview

Table of Contents

1. Goal of this project: 1

2. Using Asciidoc for creating sanity. 1

3. Steps 2

4. Examples 2

    4.1. Pieces of the puzzle 2

    4.2. Functionality 4

1. Goal of this project:

This document describes all the bits and pieces which are spread out over different issues in repositories. A lot of information is already in those issues, but this information is fragmented in such a way that there is no clear path.

Documents describing all of the points below *SHOULD* be written in such a way that deep technical knowledge about programming is not mandatory for general understanding.

Of course the deeper we get, the more detailed information will become and at a certain point it *WILL* be necessary to have this technical knowledge.

So, the goals:

1. Creating sanity
2. Consensus and understanding of the goal of the Machinekit project by clearly defining what the Machinekit project actually is about.
3. Clarity on improvements by specifying what *WILL* be improved and what *WILL NOT* be improved. A scope.
4. Insight on what parts are needed for different improvements. So a decision can be made with which parts to start, and which parts have the most benefit over different projects.
5. technical clarity on how to implement these imrovements. This means that diagrams *MUST* be made so people will know about the structural side and the technical details *WITHOUT* having to resort to dive into code to understand applications.

2. Using Asciidoc for creating sanity.

- Asciidoc is a well defined format about the structure of content.
- Asciidoctor as a program
  - is mature
  - has a big community
  - different content can be generated, for example html / pdf / epub files.
- By using asciidoctor with the asciidoctor-diagram diagrams can be made with ease in a readable format.
- Focus can thus be on creating quality content.
- We have git version control.

3. Steps

1. Gathering discussions on improvements & functional solutions. By using `asciidoctor-diagram` a document must be made giving on all pieces of the puzzle, Big and small, (im)possible solutions.
2. Define project goals and communicate these.
3. Creating links/relations between these blobs to view which parts rely on which parts.
4. Gather and define sub projects. The smaller the (sub) projects are, the easier it will be to solve them because understanding their relations with other parts of the Project is clear.
5. Subprojects *MUST* have diagrams to define the functionality we aim to have on the problems we aim to solve.
6. Subprojects *MUST* have information on the technical choices for solving the functionality.

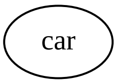
Links to various document will be put here. We want to keep things readable.

4. Examples

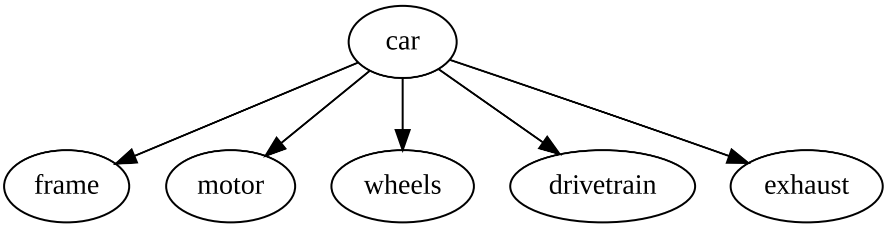
These examples are a bit silly, but they illustrate how we can get clarity and purpose in this project. This is even readable for people like me :)

4.1. Pieces of the puzzle

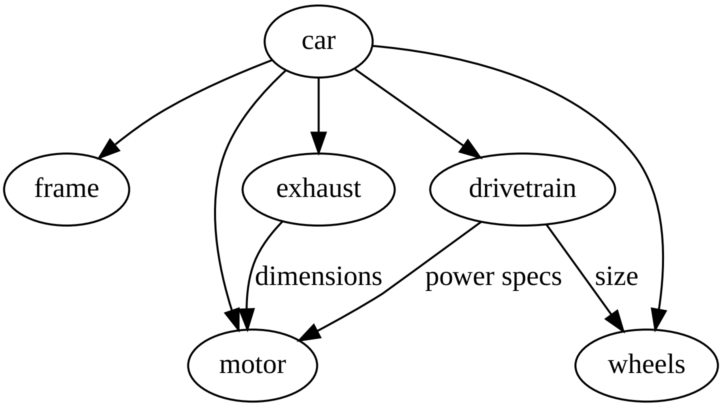
The idea is that we work top-down. Breaking problems into smaller pieces as we gather wisdom. And seeing which parts are not connected. For example, Goal: build a car.



We need to have a frame, motor, wheels, drivetrain and exhaust to make a car.

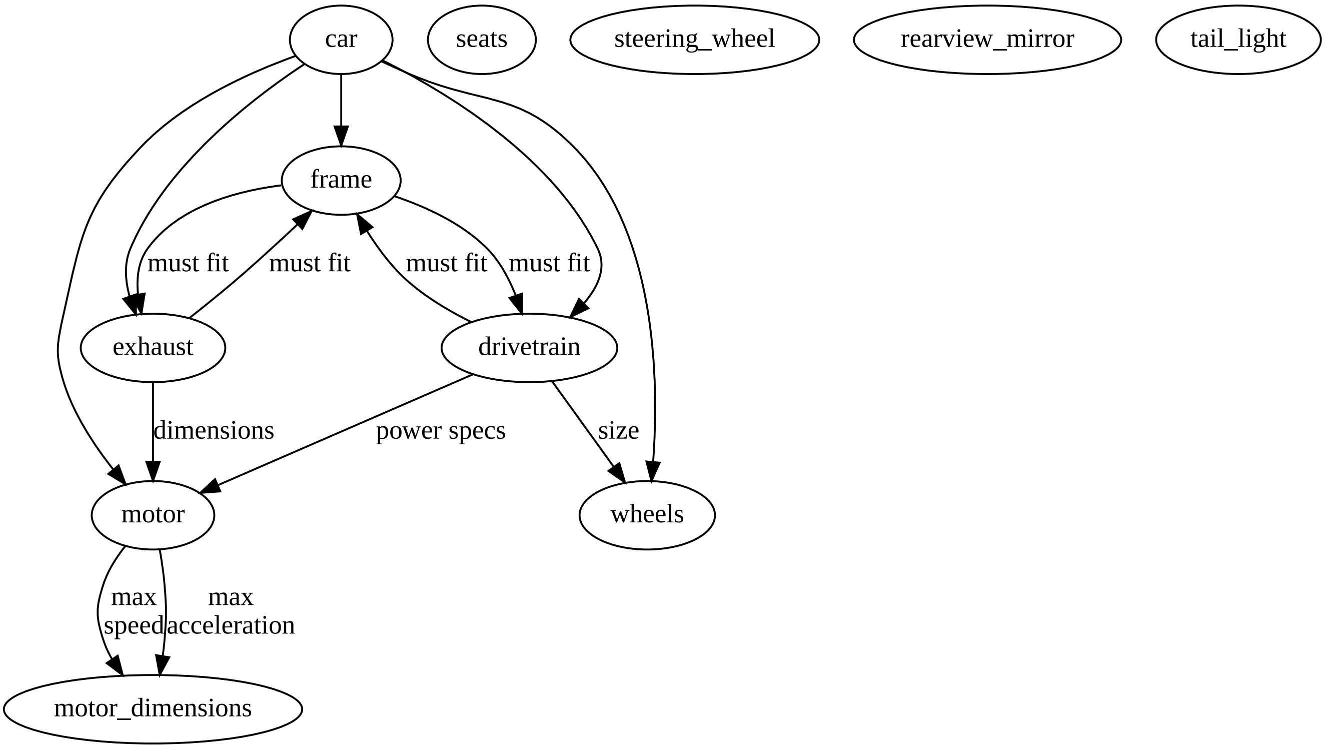


For a drivetrain we need to know about the motor specs, as well as which wheels are to be used. And for the exhaust we need to know about the motor connection too.



But wait…

Without information on desired acceleration and maximum speed we can’t pick the correct motor. And the drivetrain and exhaust must fit within the frame. And what about seats, where do they fit? rearview mirror? tail light?

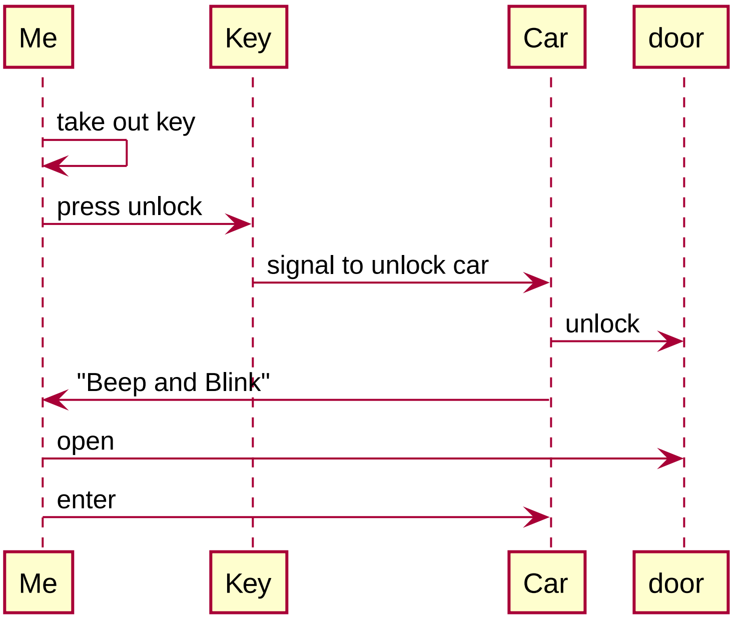


So only if we know what the relations between the pieces are we can decide where first to work on. In the same time it will also be clear *why* certain work is needed.

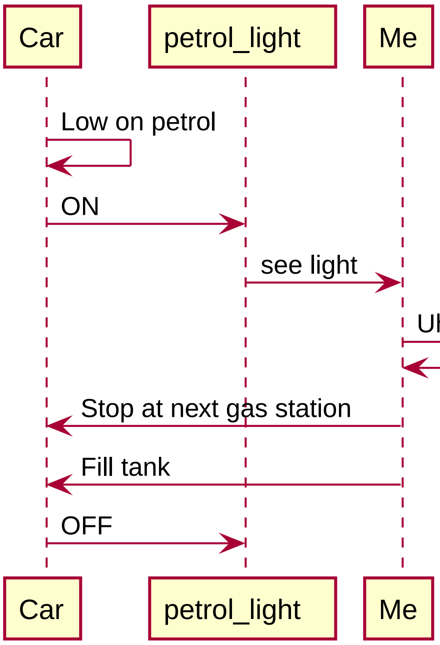
4.2. Functionality

This is a PlantUML sequence diagram on how to drive home from work.

When the work is finished, we need to open the door of the car and get seated



When driving the following can happen:


Car
petrol_light
Me
Low on petrol
ON
see light
Uh oh
Stop at next gas station
Fill tank
OFF
Car
petrol_light
Me