

Table of Contents

1. Software Architecture	2
1.1. Standpoints	2
1.2. Design	3
1.3. ROS Master	3

Welcome

Project Willy

- [History of Willy](#)
- [Project Willy](#)
- [Publicity](#)
- [Sponsors](#)

Getting started

- [Development Guide](#)
- [Driving Willy](#)
- [Documentation](#)

Build of Willy

- [Design history](#)
- [Requirements](#)
- [Design reference](#)
- [Physical build](#)
- [Hardware](#)

Robotic Operating System

- [Introduction to ROS](#)
- [ROS Tutorials](#)
- [Multi master](#)

Architecture

- [Software Architecture](#)
- [Hardware Architecture](#)
- [Skylab Architecture](#)
- [ROS topic design](#)

Hardware nodes

- [sensor node](#)
- [si node](#)
- [power node](#)
- [WillyWRT](#)

Components

- [ROS master](#)
- [New ROS master on Ubuntu](#)
- [Brain](#)

- [Sonar](#)
- [Lidar](#)
- [Localization and navigation](#)
- [Motor controller](#)
- [Joystick](#)
- [Social interaction](#)
- [Speech](#)
- [Speech recognition](#)

Skylab

- [Setup Skylab](#)
- [Python scripts](#)
- [Webserver](#)
- [Functions of the webserver](#)
- [Skylab servers](#)
- [ROS installation on Ubuntu VMs in Skylab](#)
- [DNS,DHCP, pfSense & Ubuntu](#)

Radeffect App

- [Radeffect App](#)

Lessons learned

- [Todo & Advice](#)
- [Lessons Learned](#)

Archive

- [Previous Groups](#)
- [Research Archive](#)

1. Software Architecture

1.1. Standpoints

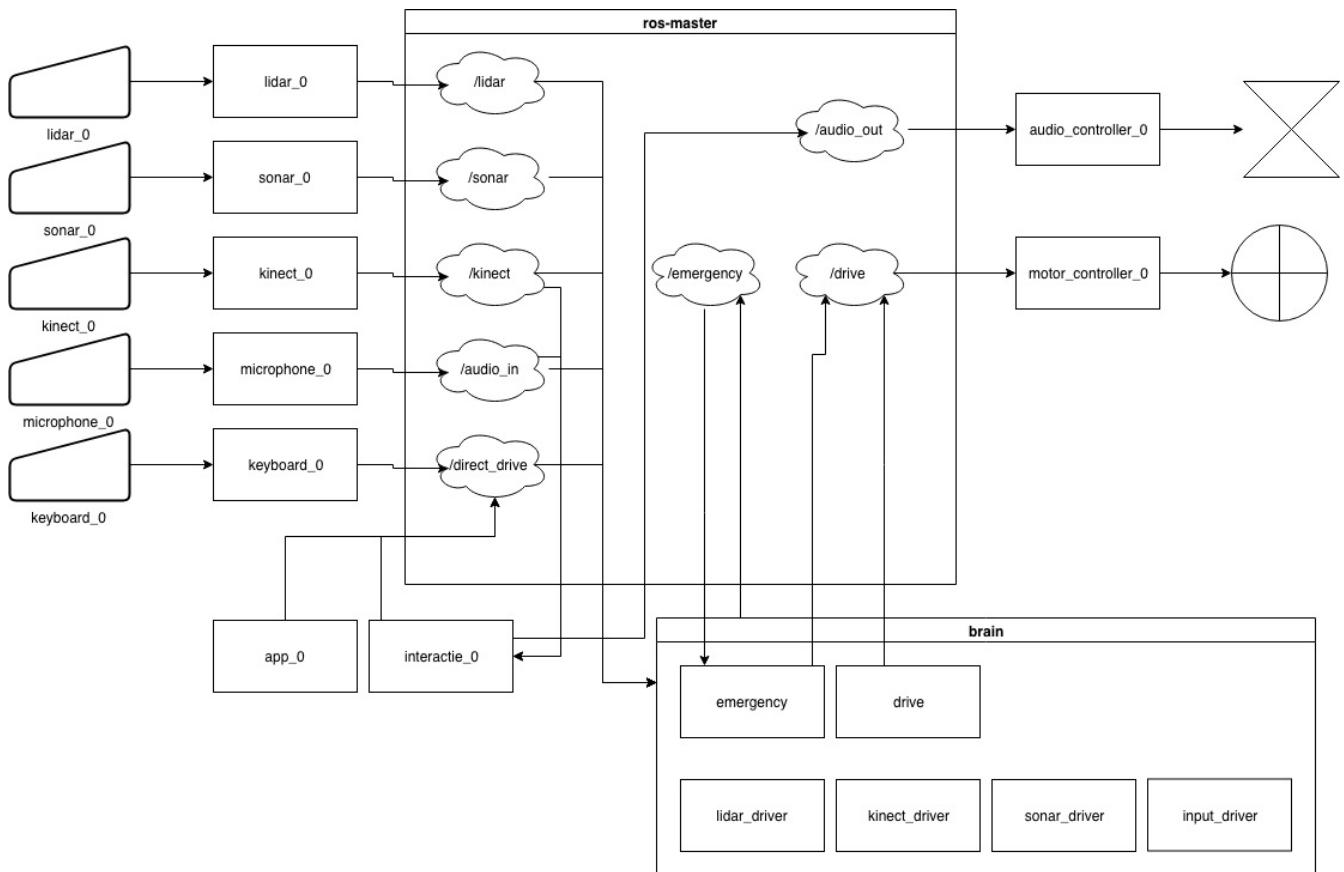
The software architecture needs to be compartmented for development puproses. To make sure the development doesn't compromise the system as a whole the following standpoints are defined to fortify the design.

- Each separte function gets a separate repository
- Each separte function has no dependency to another
- The GiT proces is applied for every repository

- Every function is running hardware independant
- Every function is running OS independant

1.2. Design

Embedding these standpoints into the Robotic Operating System (ROS) resulted in the following architure design.



1.3. ROS Master

A key feature of ROS is the topic communication. By separation of each function the ROS master, which facilitates the topics, is the key component. Every function communicate according a topic and therefore ROS master functions as a servicebus for each current and future feature.



Designing this architecture and keeping it up to date can be done by importing/editing the xml files in <https://www.draw.io/>. Draw.io is free of charge to use and requires no client.