

Robotic Operating System Podstawy

Marek Piechula
Pytech summit
08-12-22

Agenda

1. Kim jestem?
2. Co to jest ROS
3. Dlaczego ROS
4. Z czego się składa ROS
 1. Node
 2. Topic
 3. Services
 4. Parameters
 5. TF
 6. Tools do ROS2
5. Final Demo
6. Q/A

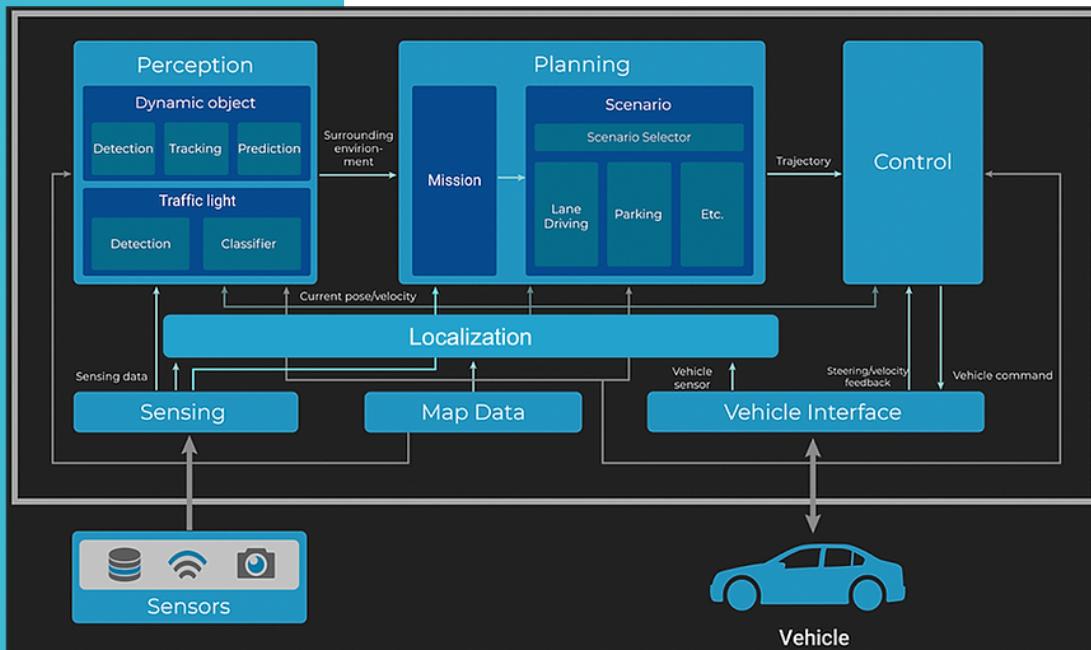
Kim jestem?



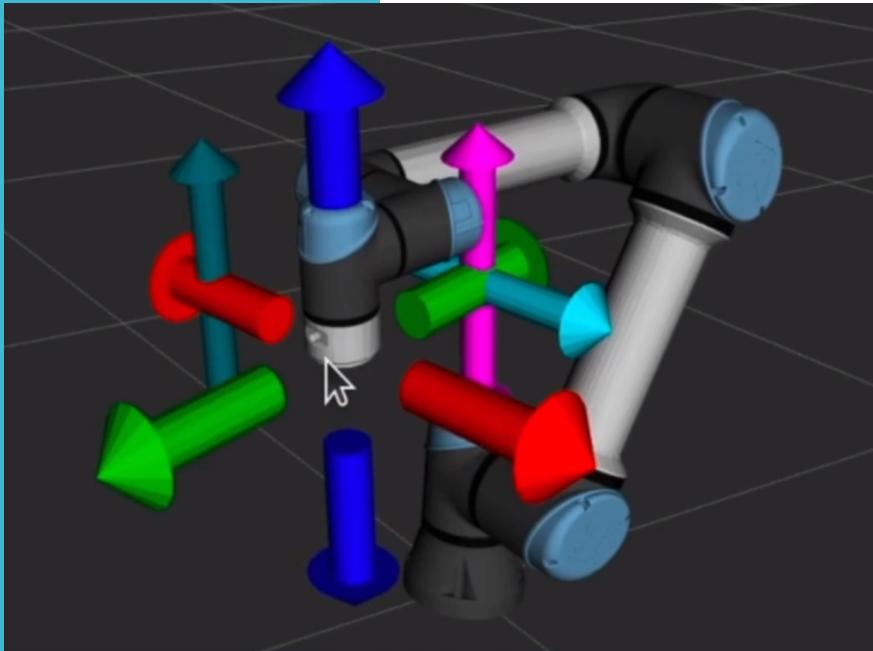
- 10 lat doświadczenia z IT
- głównie python oraz web
- od 2 lat C++ oraz automatyczne pojazdy

Moje doświadczenie z ROS:

- Pracuję w firmie Blees
- Współpracuję z Autonomous Systems
- Znam projekt Autoware.Auto – opensourcowy projekt na ROS2 dla samochodów autonomicznych



Co to jest ROS?



- **ROS** = Robot Operating System
- Jest to "framework" z zestawem bibliotek oraz narzędzi stosowanych w robotyce
- W praktyce ROS jest jak świnka morska – ani Robot, ani Operating System

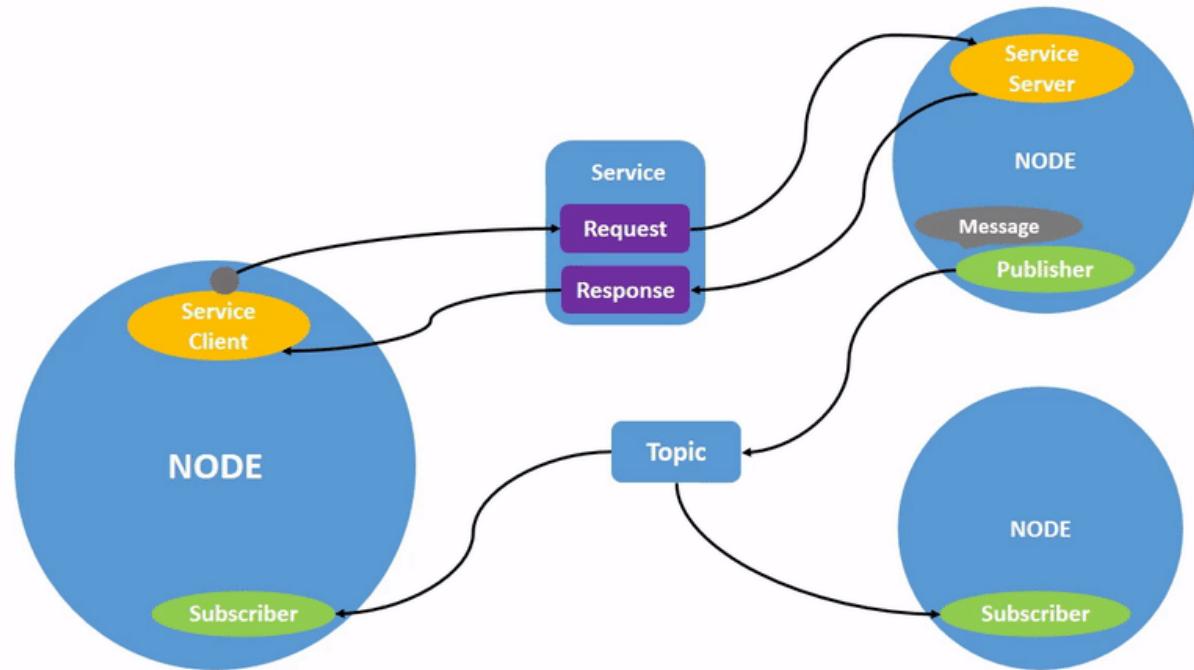


Dlaczego ROS?

- Od ponad 10 lat posiada spore community oraz jest wciąż aktywnie rozwijany
- Ma sporą liczbę narzędzi oraz sterowników do obsługi sensorów
- Wspiera C++ oraz Python (<3)
- Open Source <3
- Firmy wspierające ROS: Intel, Bosch, Microsoft, Samsung I wiele innych
- Interfejs pozwala testować swoje oprogramowanie w symulatorach (np. Carla) praktycznie jeden do jednego

Node

- Jest to niezależny moduł ("cegielka") w systemie ROS który odbiera oraz przesyła do innych węzłów dane.
- Przykładem węzła mogą być sensory; kontroler sterujący czy planowanie trasy



Topic

- Jest to "adres" gdzie węzeł może odbierać dane lub przesyłać
- Np. węzeł GNSS wysyła lokalizację do topicu, który jest odbierany przez węzeł kontrolujący
- Sam Topic jest kolejką
- Sama obsługa kolejki jest definiowana przez QoS (Quality of Service)
- Komunikacja w ROS2 jest zapewniona przez DDS (Data Distribution Service)

Service

- Jest bardzo podobny do Topic
- Jego różnica jest taka, że po wysłaniu wiadomości, wysyłający węzeł będzie czekać na odpowiedź
- Np. akcja „zamknij drzwi” – powiodła się czy nie?

Parameters

- Każdy węzeł może mieć dodatkowe parametry
- Najczęściej parametry są wykorzystywane do ustawienia węzła przed jego uruchomieniem
- Mogą być parametry ustawiane dynamicznie

TF

- TF to rozproszony system do budowania drzewa transformacji poszczególnych połączeń w czasie
- Np. Pojazd ma transformację względem mapy oraz koła mają transformację względem pojazdu
- Z TF jest powiązany URDF który opisuje model "robota"
- Działa to naprawdę dobrze :-)

Toole do ROS2

- RVIZ
- Plotjuggler
- Rosbag
- Colcon
- ROS CLI
- RQT
- Robot localization

Final Demo

- Kod w pythonie z "AI" która zatrzymuje się przed przeszkodami
- Wykorzystuje odbieranie danych o przeszkodach, transformację względem pojazdu a przeszkód oraz wysyłanie o szybkości

Sumarum

- Węzły to niezależne komponenty które są komunikują z innymi węzłami
- Węzły mogą wysyłać wiadomości do topiców oraz odbierać
- Węzły też posiadają serwisy do których wysyła się żądanie (request) i jest wysyłana odpowiedź (response)
- Węzły posiadają też parametry które mogą być ustalane dynamicznie
- TF to rozproszony system połączeń w czasie

Linki

- <https://vimeo.com/146183080>
- <https://design.ros2.org>
- <https://docs.ros.org/en/humble/Concepts.html>
- <https://docs.ros.org/en/humble/Tutorials.html>
- <https://github.com/firemark/ros-basic-prelection>
- <http://wiki.ros.org/rviz>
- <https://www.plotjuggler.io>

Q/A