

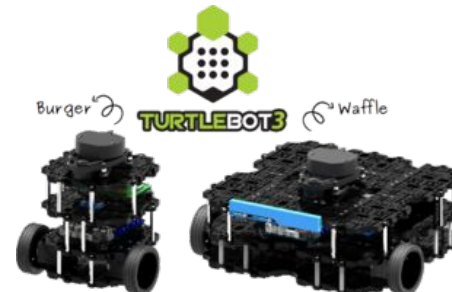
ROS and Gazebo

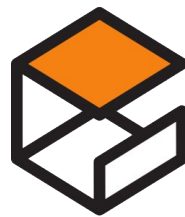
Matt Taylor, Ojas Upalekar, and Evan Yu



Overview

- Gazebo
 - Definition
 - Requirements
- ROS
 - Definition
 - Capabilities
- ROS 2
 - Key Differences
 - NAV 2
- Current Project
 - Goal
 - TurtleBot3
 - Current Progress
 - Future Work





Gazebo

- Created by Open Robotics
- Open source 3D simulation created to simulate robots in complex indoor and outdoor environments.
- Currently supported by Ubuntu and Mac and not fully functional on Windows.
- Computer must have a dedicated GPU



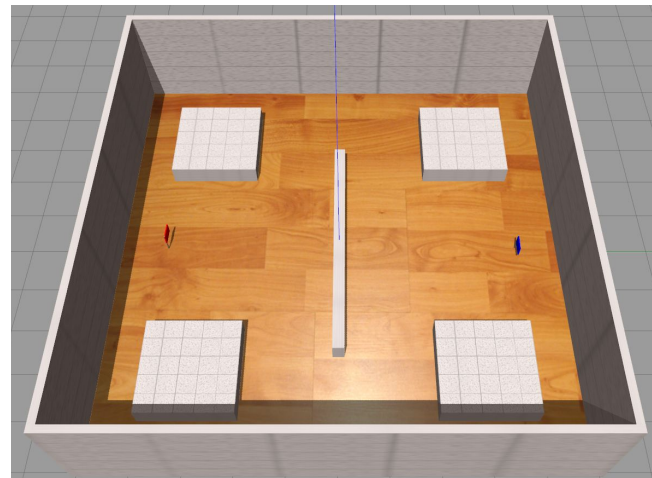
<https://gazebosim.org/home>





Gazebo Requirements

- **gzserver**
 - Executable that runs the physics update-loop and sensor data generation
- **gzclient**
 - Executable that runs the GUI
- **World File**
 - SDF file to describe the simulation contents
 - Can use existing worlds or create custom builds
- **Launch File**
 - Main file to launch world, robot, gzserver, and gzclient
 - Can be xml format (ROS) or a python file (ROS 2)



Custom CtF World File - "easy_ctf.world"





Gazebo World Example

```
<model name='middle_wall'>
  <pose>0 0 0 0 -0 0</pose>
  <scale>1 1 1</scale>
  <link name='Wall_12'>
    <pose>0 0 0 0 -0 3.14159</pose>
    <velocity>0 0 0 0 -0 0</velocity>
    <acceleration>0 0 0 0 -0 0</acceleration>
    <wrench>0 0 0 0 -0 0</wrench>
  </link>
</model>
```



Middle Wall

```
<light name='sun' type='directional'>
  <cast_shadows>1</cast_shadows>
  <pose>0 0 10 0 -0 0</pose>
  <diffuse>0.8 0.8 0.8 1</diffuse>
  <specular>0.2 0.2 0.2 1</specular>
  <attenuation>
    <range>1000</range>
    <constant>0.9</constant>
    <linear>0.01</linear>
    <quadratic>0.001</quadratic>
  </attenuation>
  <direction>-0.5 0.1 -0.9</direction>
  <spot>
    <inner_angle>0</inner_angle>
    <outer_angle>0</outer_angle>
    <falloff>0</falloff>
  </spot>
</light>
```

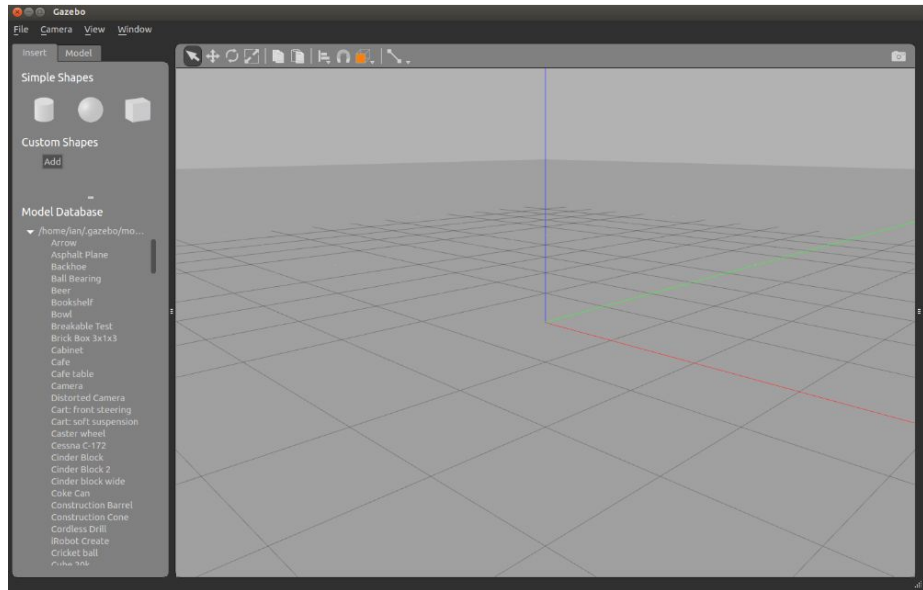
Sun and Shadow Rendering





Gazebo Model Editor

- Create worlds with a GUI instead of using SDF format
- Automatically converts rendering into a world (SDF) file for the user
- This is how “easy_ctf.world” was created





Simple Launch File Example

- Python launch files require a `generate_launch_description()` method which returns a `LaunchDescription` object.
- A launch description object contains a list of actions.
- [Actions](#) can be worlds, other launch files, nodes, and much more.

```
import os

from ament_index_python.packages import get_package_share_directory
from launch import LaunchDescription
from launch.actions import DeclareLaunchArgument
from launch.actions import IncludeLaunchDescription
from launch.conditions import IfCondition
from launch.launch_description_sources import PythonLaunchDescriptionSource
from launch.substitutions import LaunchConfiguration
from launch_ros.actions import Node

def generate_launch_description():

    pkg_gazebo_ros = get_package_share_directory('gazebo_ros')
    pkg = get_package_share_directory('ctf_package')

    # Gazebo launch
    gazebo = IncludeLaunchDescription(
        PythonLaunchDescriptionSource(
            os.path.join(pkg_gazebo_ros, 'launch', 'gazebo.launch.py'),
        )
    )

    return LaunchDescription([
        DeclareLaunchArgument(
            'world',
            default_value=[os.path.join(pkg, 'worlds/empty_worlds', 'empty.world'), ''],
            description='SDF world file'),
        gazebo
    ])
```

Empty World Launch File





ROS - Robot Operating System

- Created by Open Robotics
- Not an operating system, but middleware to control robots.
- Allows package creation to modularize functionalities
- Released as distributions or distros.
 - ROS1: Kinetic, Noetic
 - ROS2: Humble, Foxy, Galactic

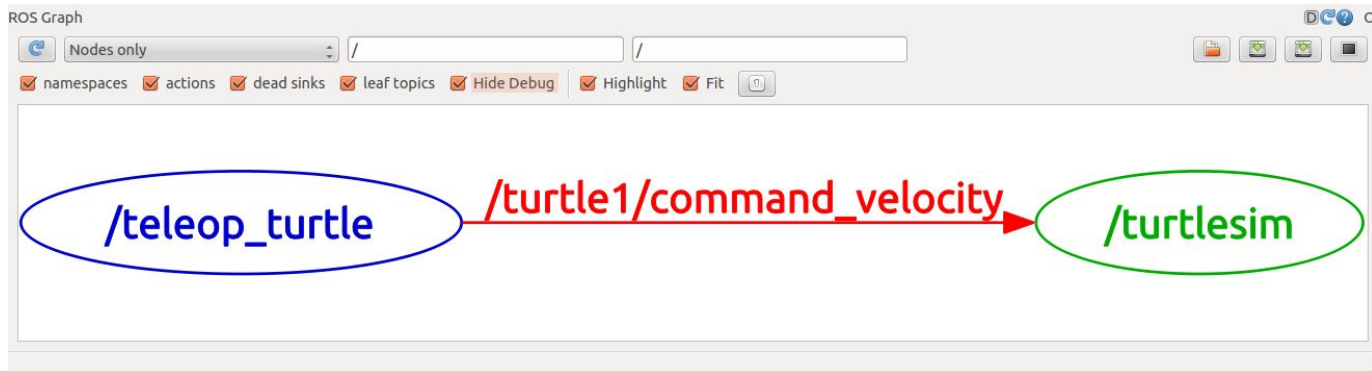


<https://ubuntu.com/engage/ros-kinetic-eol>



Packages, Nodes and Topics

- Packages: Contains nodes, datasets, configuration files
 - Intended to provide functionality in an easy-to-consume manner so that software can be easily reused
- Nodes: An executable program or process that performs computation.
 - roscout: Collects and logs nodes' debugging output.
- Topics: Named buses over which nodes exchange messages
 - /cmd_vel: Provides messages for working with geometric primitives: points, vectors, and poses.



Why ROS 2

- ROS1 End of Life in 2025
- Improved Node Creation
- Flexible Launch files
- Easier integration of multiple packages
- Can be used on Ubuntu, MacOS, and Windows 10



<https://docs.ros.org/en/foxy/index.html>





Navigation2 (NAV2)

- ROS2 version of the Navigation Stack
 - `move_base`: ROS1 version
- Relies on behavior trees to have more complex movement such as waypoint navigation

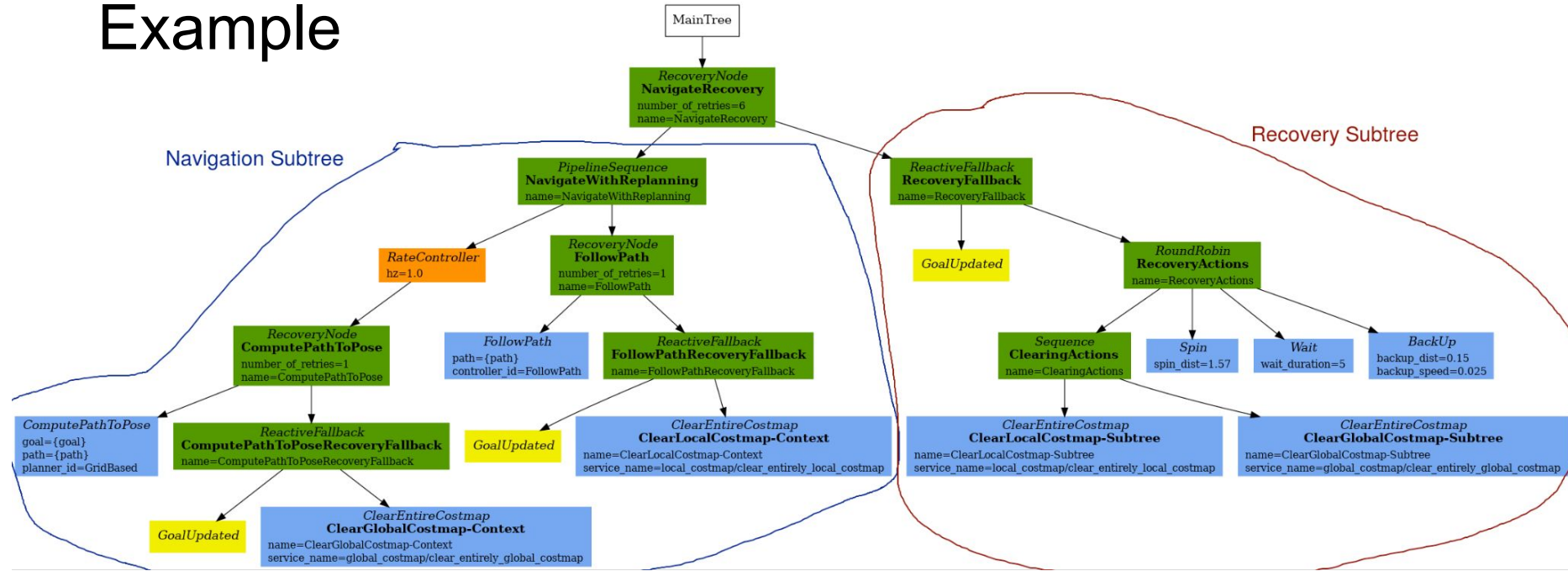


N A V 2

<https://navigation.ros.org/index.html>

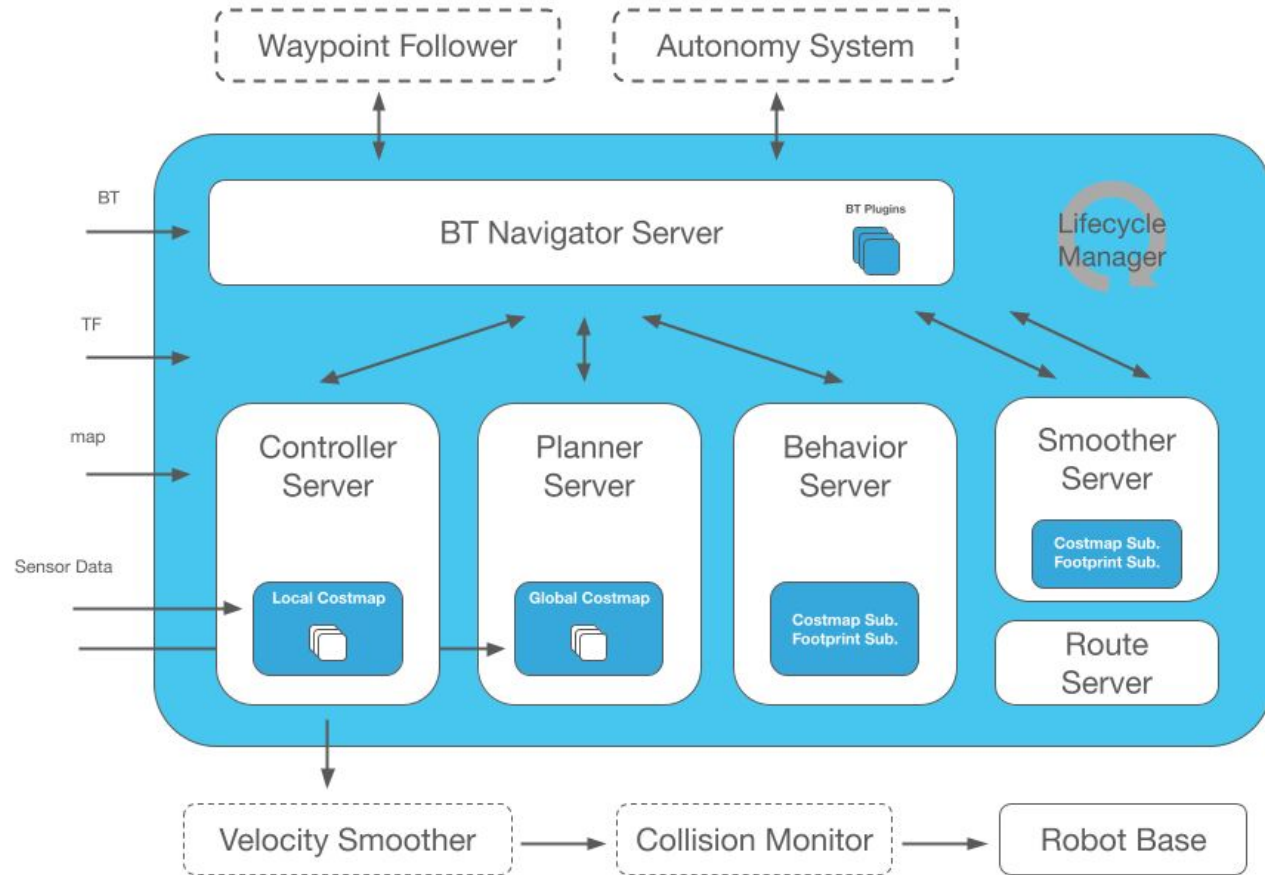


Navigation and Recovery Behavior Tree Example



https://navigation.ros.org/behavior_trees/overview/detailed_behavior_tree_walkthrough.html



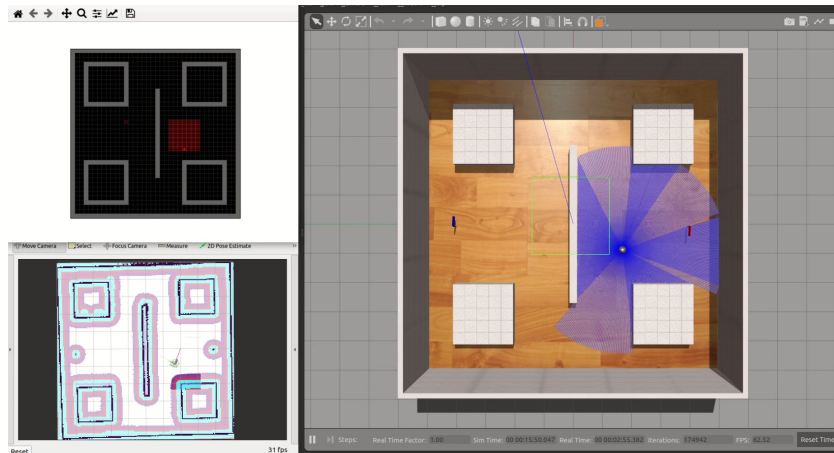




Update on Current Project

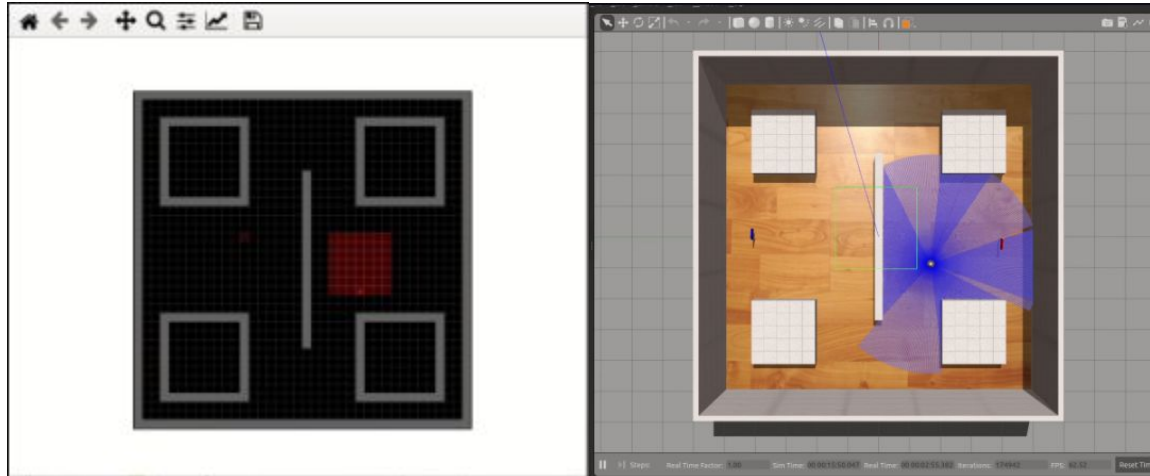
Current status:

- CTF world with two TurtleBot3 agents
 - Motion only works for 1 agent
 - Single agent navigates between waypoints using Nav2
 - Gazebo > Python/MultiGrid > Motion policy > ROS2/NAV2 > Gazebo



MultiGrid Integration

- Custom grid layout created for our CTF map
- Provides approximation of location of robot in Gazebo to an RL model
- Currently only has one robot, but can be expanded to include more





Code structure

ROS2

- NAV2 package allows the robot to move between waypoints

MultiGrid

- Simple gridworld representation of CTF map generated and maintained by MultiGrid
- Gets robot state from Gazebo and updates state before sending to motion policy

Gazebo

- Physically realistic simulation of robot motion, takes robot motion commands from ROS2

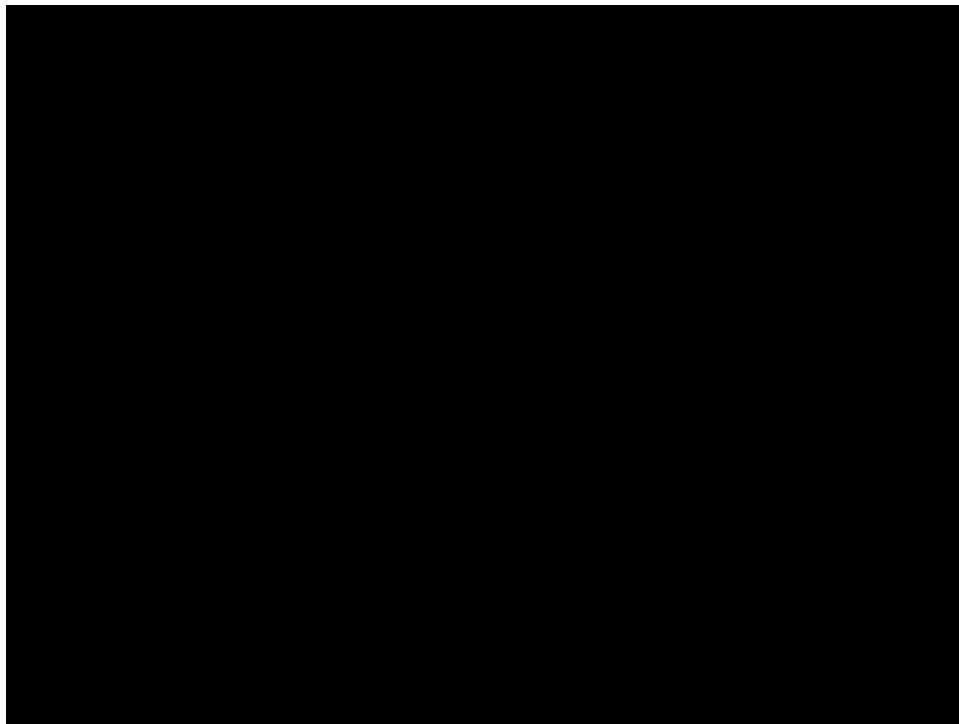
RViz

- Not part of the code, but a nice feature of ROS2 that gives a visualization of the map and the path the robot is taking/will take





CTF Demo Video





Future Work

- Multi-robot integration
 - Spawn and move two robots
 - Multi-robot synchronization
- Episode end condition from multigrid
- More advanced decision policy
- Model CTF engagements in Gazebo
- Hardware-in-the-Loop testing with TurtleBot3

