Turtlebot4 Nav2

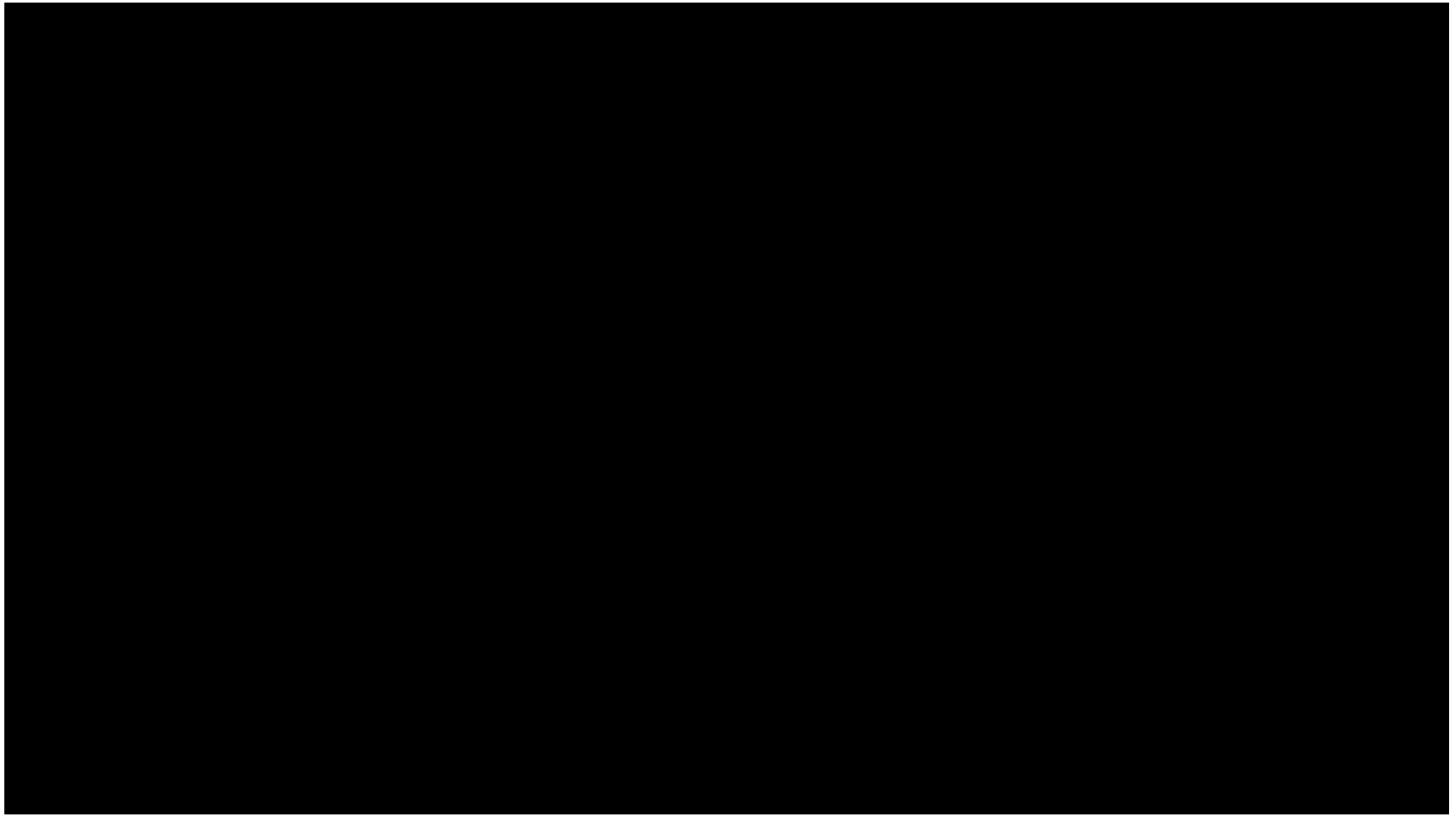
운영체제의 실제

안인규 (Inkyu An)

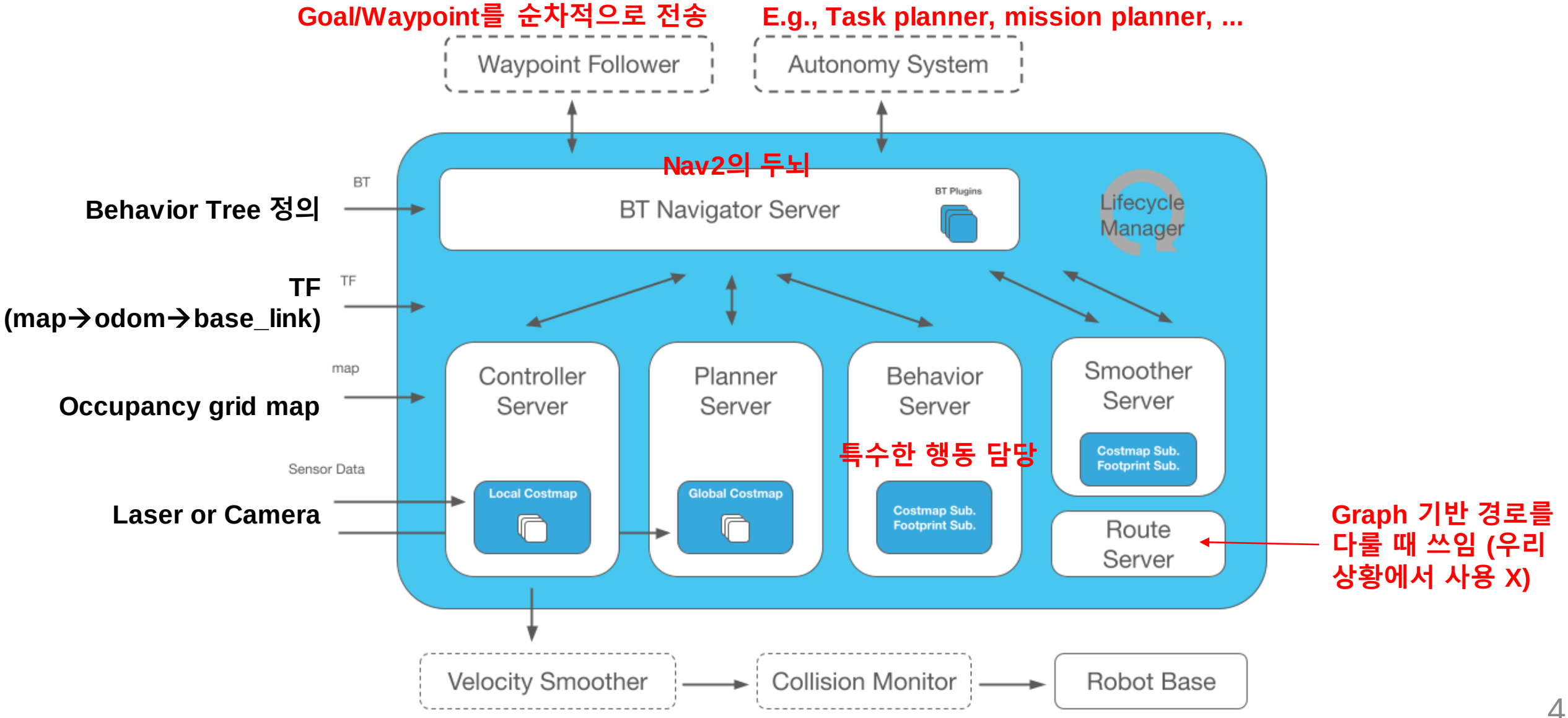
Navigation2 Overview

- Autonomous Vehicle 기술을 모바일 및 서비스 로봇에 맞게 다운사이징 및 최적화한 프레임워크
  - Autonomous movement for a robot in a 2D map
  - Given a ‘current pose’ and a ‘goal pose’, Path is panned and Robot drives itself to the goal
- One of the key and most used packages of ROS

Navigation2 Overview



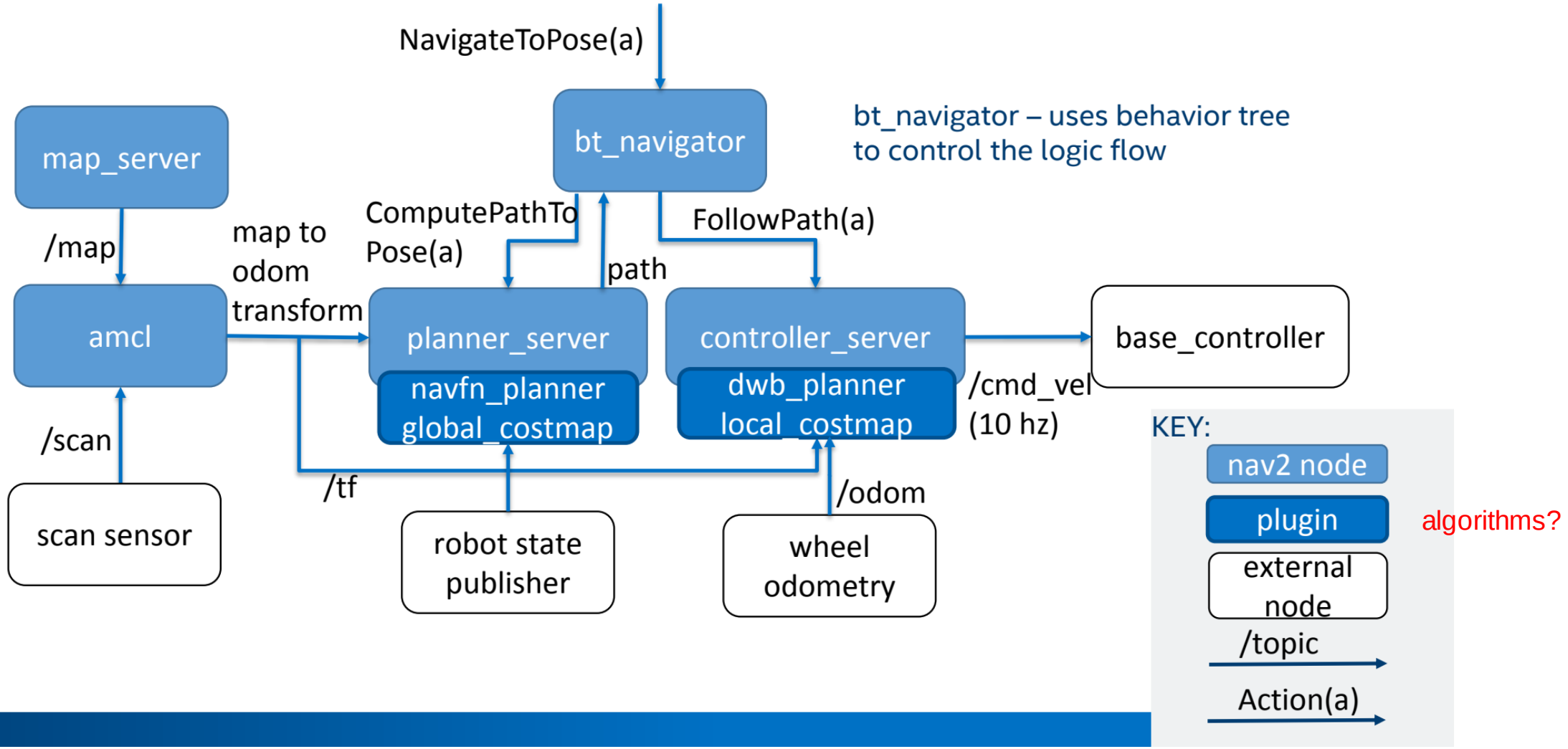
Navigation2 Overview



Navigation2 Overview

- **BT Navigator <-> 각 Server**: Action 기반 통신
  - BT Navigator → Planner Server
    - ComputePathToPose: 현재 Pose에서 Goal까지 global path 계산
  - BT Navigator → Controller Server
    - FollowPath: Path를 따라가며 cmd_vel 생성
  - BT Navigator → Behavior Server
    - Backup: 로봇을 일정 거리만큼 뒤로 이동
    - Spin: 제자리에서 지정된 각도 회전
    - Wait: 지정된 시간 동안 정지
    - …
  - BT Navigator → Smoother Server
    - SmoothPath: Planner가 만든 path 부드럽게 생성

Navigation2 Overview

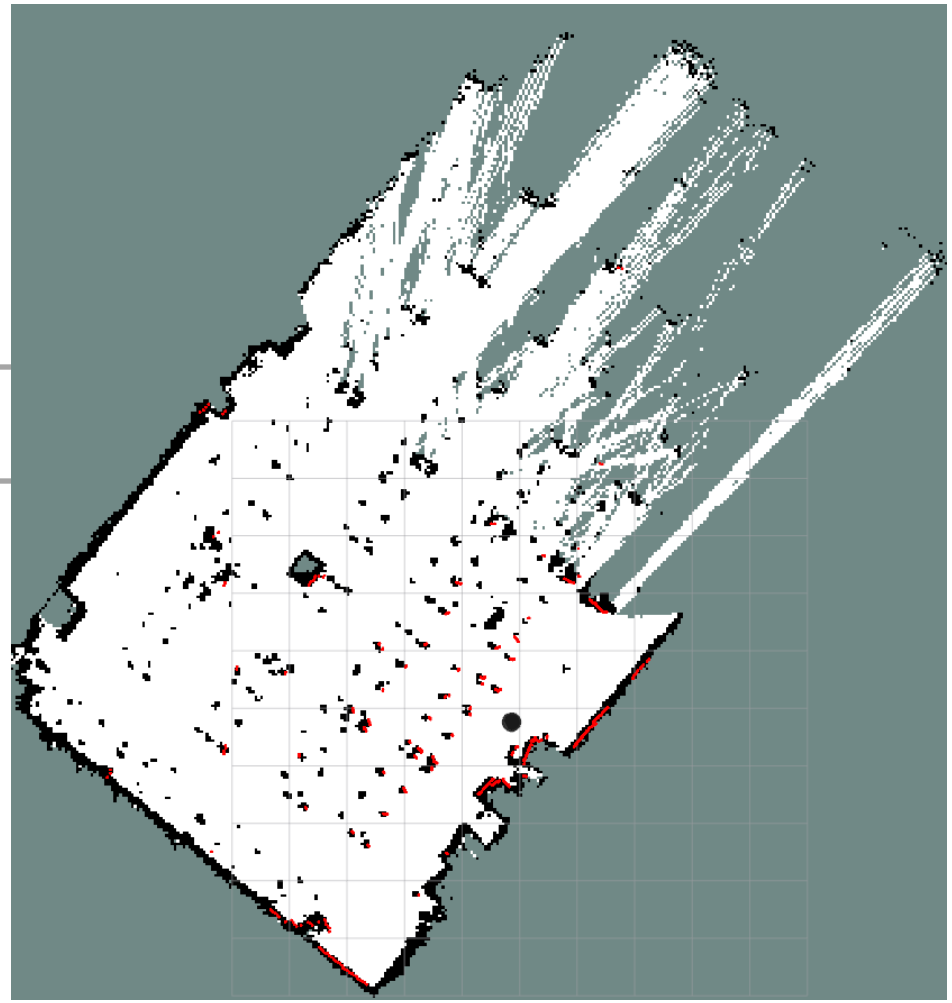


Imaged from Intel, Matt Hansen's PDF

Turtlebot4 – Nav2 (Mapping)

- Run SLAM (recommended to run synchronous SLAM on a remote PC to get a higher resolution map):

```
$ ros2 launch turtlebot4_navigation slam.launch.py
```



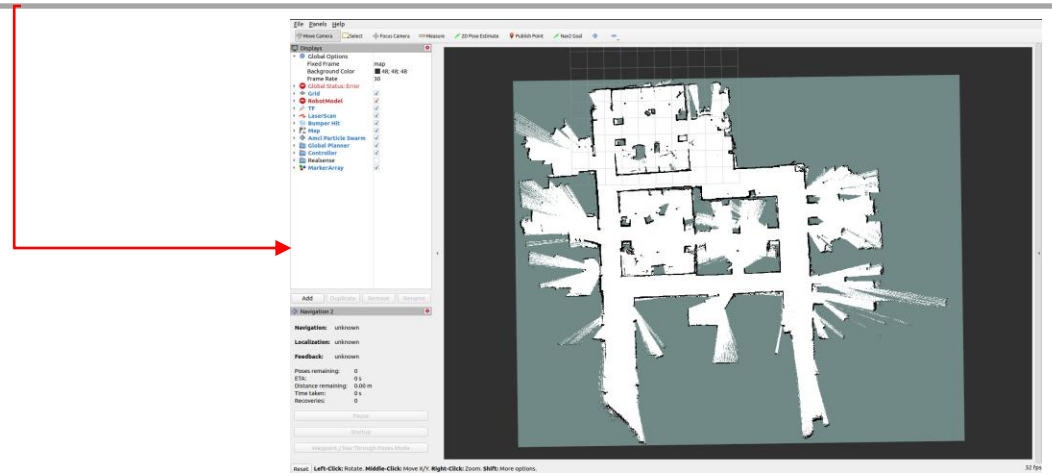
Turtlebot4 – Nav2 (Navigating)

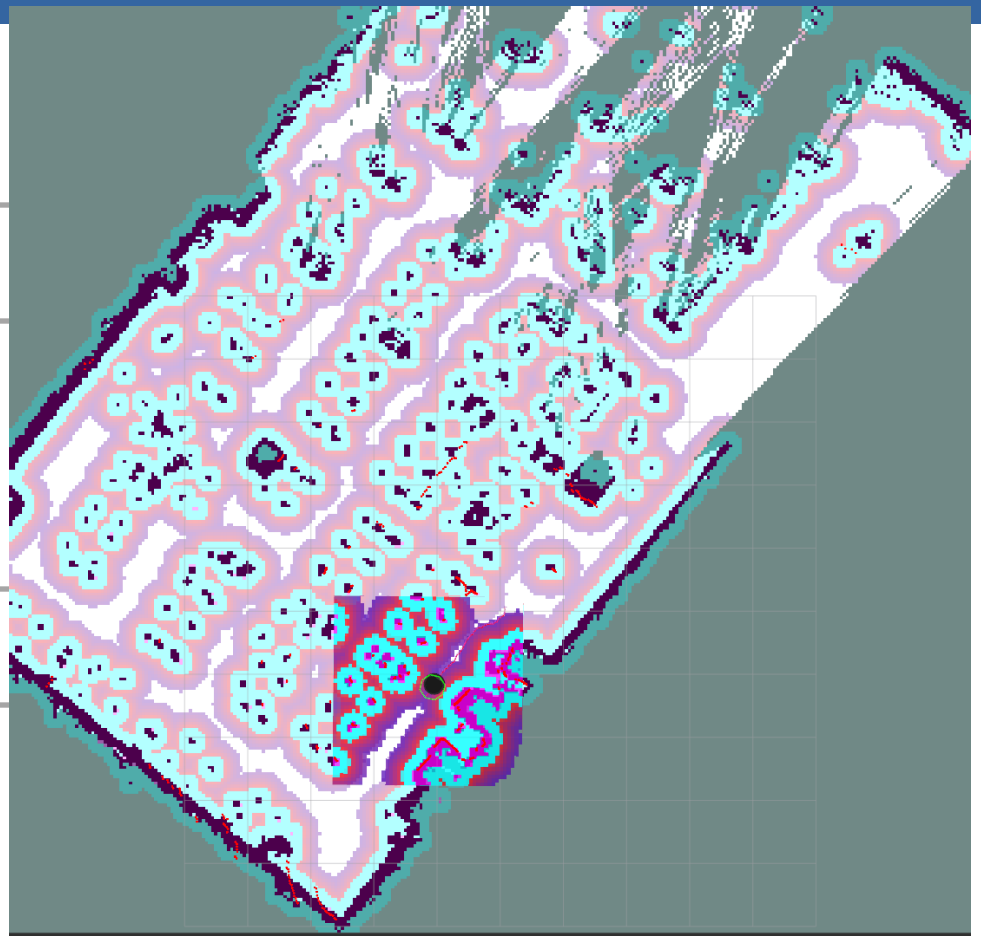
- Launch nav2:

```
$ ros2 launch turtlebot4_navigation nav2.launch.py
```

- View the map and interact with navigation:

```
$ ros2 launch turtlebot4_viz view_navigation.launch.py
```





Turtlebot4 – Nav2 (Set Goal)

- The Nav2 Goal tool allows you to set a goal pose for the robot (topic name: /goal_pose)