RTAB 3D mapping and navigation

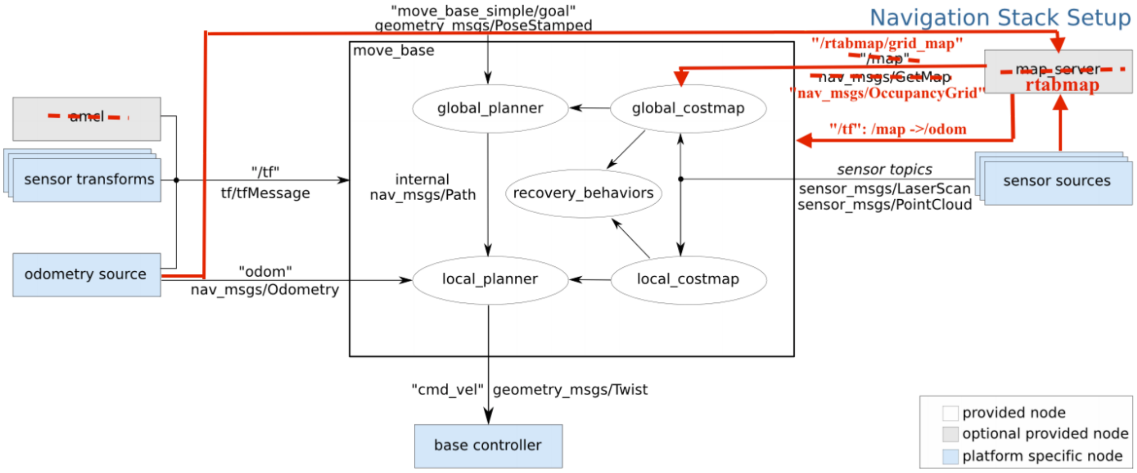
The following workspace contains the function packages in the entire rplidar_ws. If you need to transplant it to your own development board, you need to copy all the function packages to the src of the workspace for compilation, and install the relevant environment.

Note: This course uses Rosmaster-X3Plus as an example. It needs to be run with a depth camera. Rosmaster-X3 is run with an Astrapro depth camera.

Function package path: ~/ydlidar_ws/src/yahboomcar_nav

1. Introduction

This software package is the ROS function package of RTAB Map, which is an RGB-D SLAM method based on a global loop closure detector with real-time constraints. The software package can be used to generate a three-dimensional point cloud of the environment and create a two-dimensional occupancy grid map for navigation.



As can be seen from the above figure, Monte Carlo positioning amcl is not required. RTAB Map has its own positioning function; if used, it will cause repeated positioning and positioning failure. When using the RTAB Map navigation core framework, the initialized map is provided by RTAB Map, not map_server.

2. Mapping and Use

Input following command:

```
roslaunch yahboomcar_nav laser_astrapro_bringup.launch
roslaunch yahboomcar_nav yahboomcar_rtabmap.launch use_rviz:=true
```

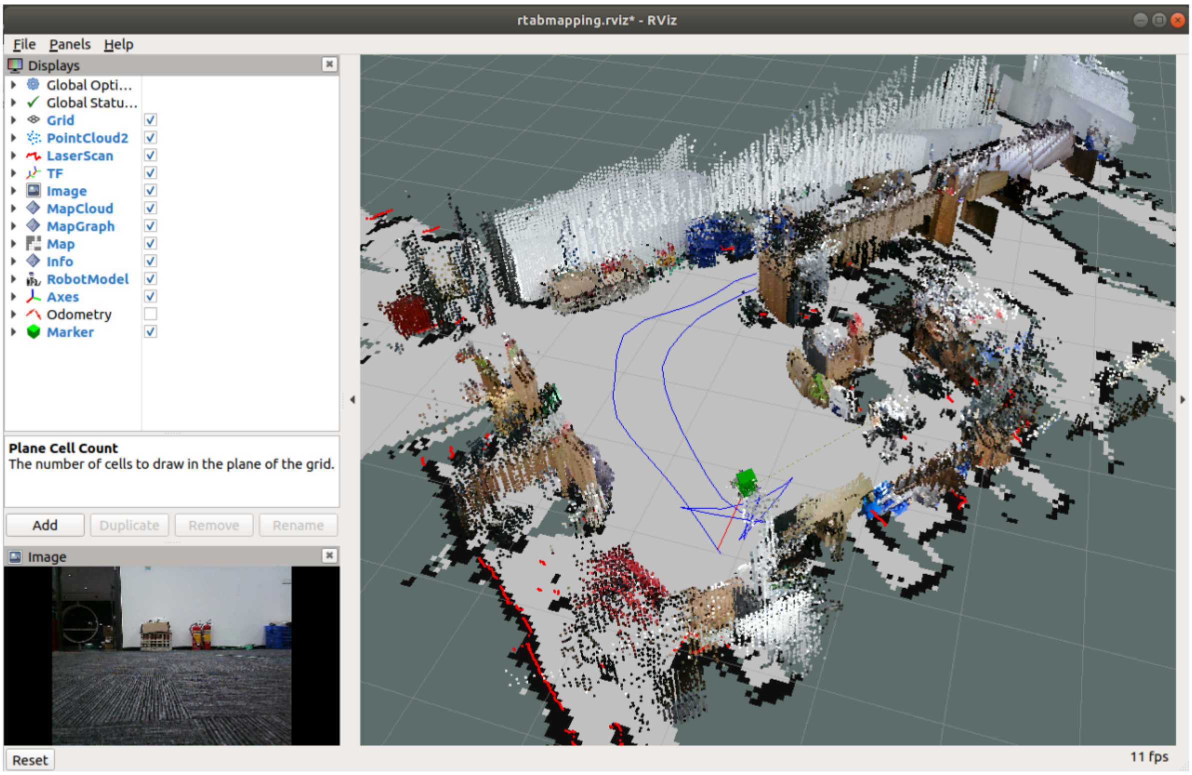
- use_rviz parameter: whether to open rviz.

Keyboard control car:

```
roslaunch yahboomcar_ctrl yahboom_keyboard.launch
```

After starting according to the above method, choose any one method to control the mapping;

The slower the speed when constructing the map, the better the effect (especially the angular speed); the robot fills the area to be mapped and the map is as closed as possible.



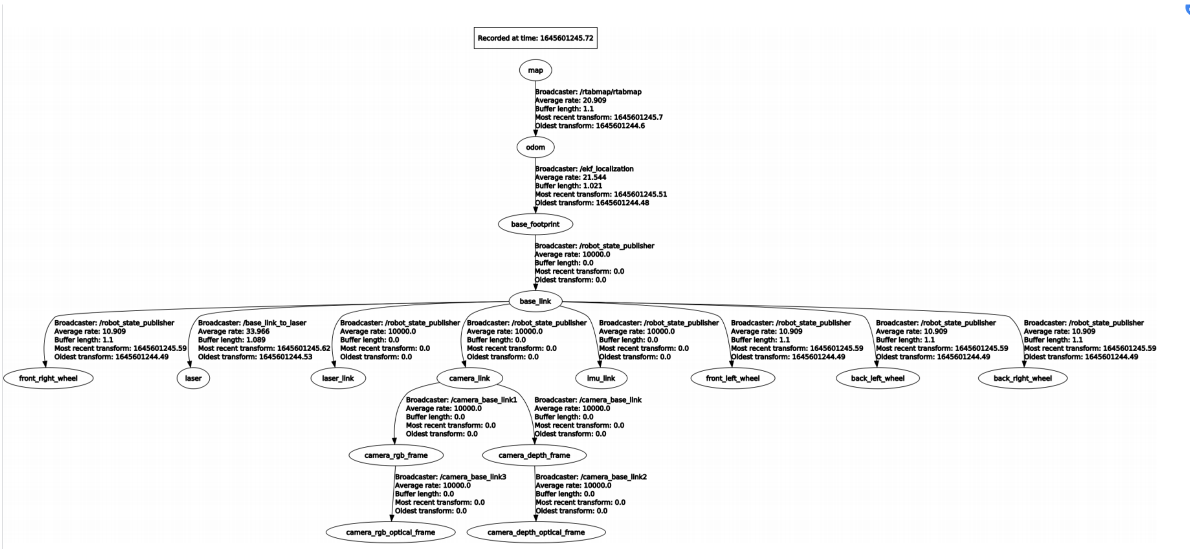
Plane Cell Count
The number of cells to draw in the plane of the grid.

 When the map is completed, directly ctrl+c to exit the map node, the system will automatically save the map.

The default save path of the map is [~/.ros/rtabmap.db].

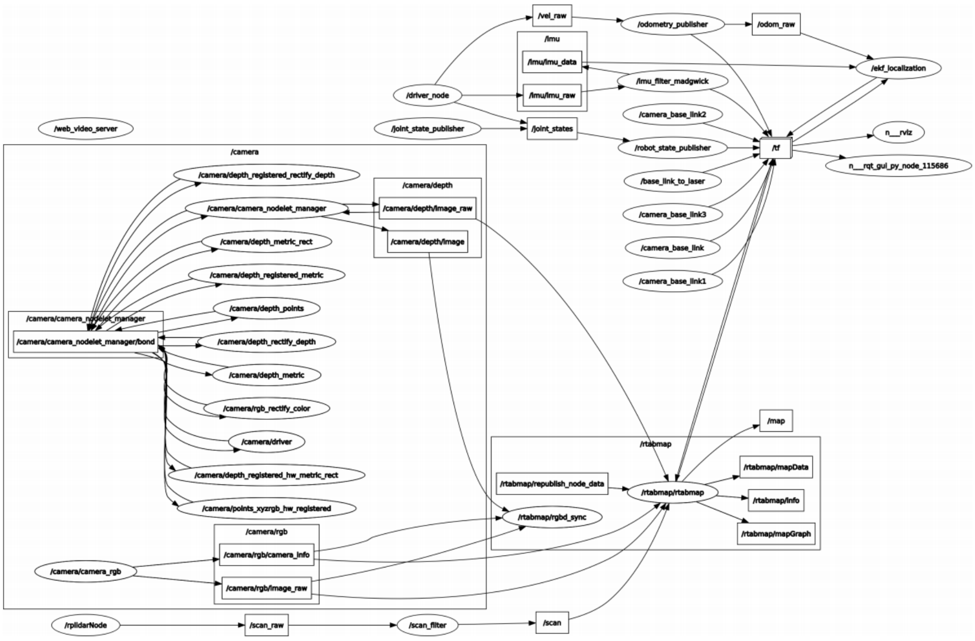
View tf tree:

```
rosrun rqt_tf_tree rqt_tf_tree
```


Recorded at time: 1645601245.72
map
Broadcaster: /rtabmap/rtabmap
Average rate: 20.909
Buffer length: 1.1
Most recent transform: 1645601245.7
Oldest transform: 1645601244.6
odom
Broadcaster: /ekf_localization
Average rate: 21.544
Buffer length: 1.021
Most recent transform: 1645601245.51
Oldest transform: 1645601244.48
base_footprint
Broadcaster: /robot_state_publisher
Average rate: 10000.0
Buffer length: 0.0
Most recent transform: 0.0
Oldest transform: 0.0
base_link
Broadcaster: /robot_state_publisher
Average rate: 10.909
Buffer length: 1.1
Most recent transform: 1645601245.59
Oldest transform: 1645601244.49
front_right_wheel
Broadcaster: /base_link_to_laser
Average rate: 33.966
Buffer length: 1.089
Most recent transform: 1645601245.62
Oldest transform: 1645601244.53
laser
Broadcaster: /robot_state_publisher
Average rate: 10000.0
Buffer length: 0.0
Most recent transform: 0.0
Oldest transform: 0.0
laser_link
Broadcaster: /robot_state_publisher
Average rate: 10000.0
Buffer length: 0.0
Most recent transform: 0.0
Oldest transform: 0.0
camera_link
Broadcaster: /robot_state_publisher
Average rate: 10000.0
Buffer length: 0.0
Most recent transform: 0.0
Oldest transform: 0.0
imu_link
Broadcaster: /robot_state_publisher
Average rate: 10.909
Buffer length: 1.1
Most recent transform: 1645601245.59
Oldest transform: 1645601244.49
front_left_wheel
Broadcaster: /robot_state_publisher
Average rate: 10.909
Buffer length: 1.1
Most recent transform: 1645601245.59
Oldest transform: 1645601244.49
back_left_wheel
Broadcaster: /robot_state_publisher
Average rate: 10.909
Buffer length: 1.1
Most recent transform: 1645601245.59
Oldest transform: 1645601244.49
back_right_wheel
Broadcaster: /camera_base_link1
Average rate: 10000.0
Buffer length: 0.0
Most recent transform: 0.0
Oldest transform: 0.0
camera_rgb_frame
Broadcaster: /camera_base_link
Average rate: 10000.0
Buffer length: 0.0
Most recent transform: 0.0
Oldest transform: 0.0
camera_depth_frame
Broadcaster: /camera_base_link3
Average rate: 10000.0
Buffer length: 0.0
Most recent transform: 0.0
Oldest transform: 0.0
camera_rgb_optical_frame
Broadcaster: /camera_base_link2
Average rate: 10000.0
Buffer length: 0.0
Most recent transform: 0.0
Oldest transform: 0.0
camera_depth_optical_frame

View node:

```
rqt_graph
```



As can be seen from the above figure, the information that the [rtabmap] node needs to subscribe to: radar data, camera data, and tf data.

3. Navigation and obstacle avoidance

Input the following command to start the underlying driver:

```
roslaunch yahboomcar_nav laser_astrapro_bringup.launch
```

Input the following command to start mapping or navigation:

```
roslaunch yahboomcar_nav yahboomcar_rtabmap_nav.launch use_rviz:=False
```

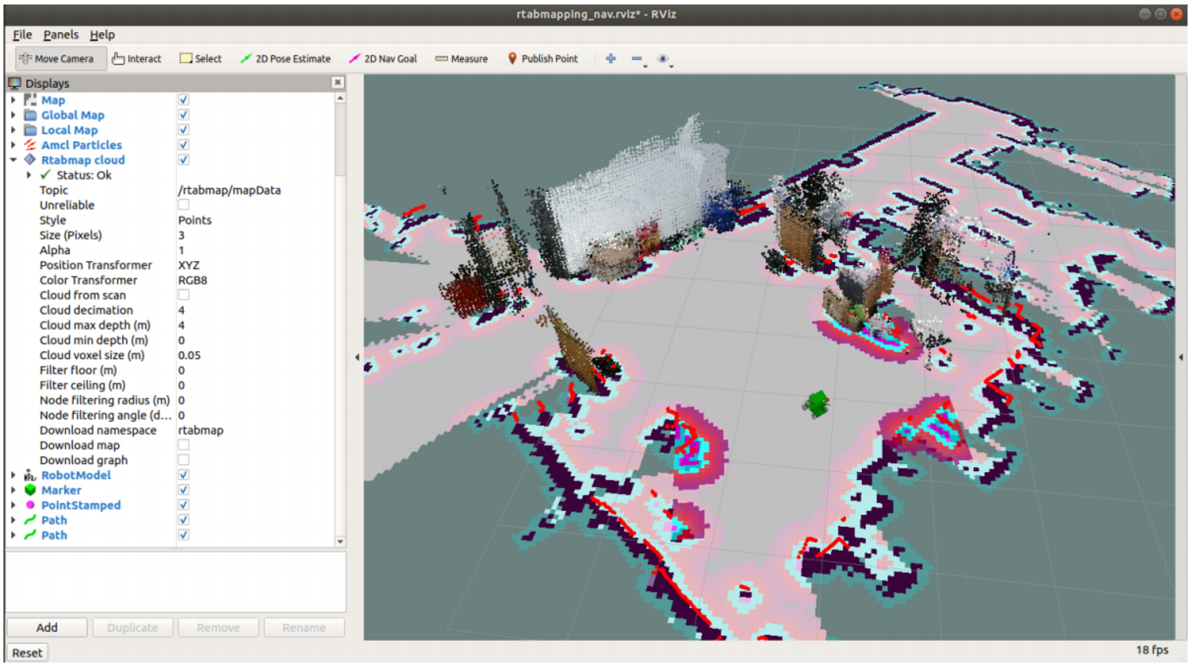
- use_rviz parameter: whether to open rviz.

Keyboard control node:

```
roslaunch yahboomcar_ctrl yahboom_keyboard.launch
```

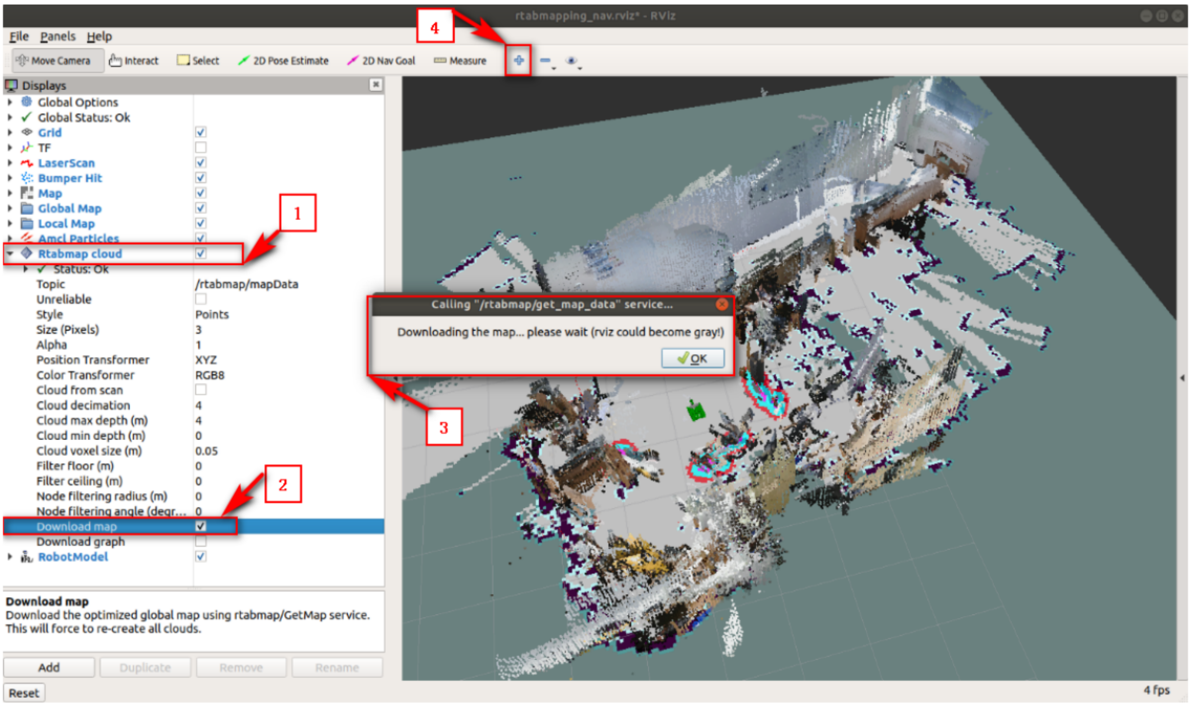
When the navigation mode is turned on, the system automatically loads the 2D raster map.

It cannot directly load the 3D map and needs to be loaded manually.



- Map ☑
- Global Map ☑
- Local Map ☑
- Amcl Particles ☑
- Rtabmap cloud ☑
  - Status: Ok
  - Topic /rtabmap/mapData
  - Unreliable □
  - Style Points
  - Size (Pixels) 3
  - Alpha 1
  - Position Transformer XYZ
  - Color Transformer RGB8
  - Cloud from scan □
  - Cloud decimation 4
  - Cloud max depth (m) 4
  - Cloud min depth (m) 0
  - Cloud voxel size (m) 0.05
  - Filter floor (m) 0
  - Filter ceiling (m) 0
  - Node filtering radius (m) 0
  - Node filtering angle (d... 0
  - Download namespace rtabmap
  - Download map □
  - Download graph □
- RobotModel ☑
- Marker ☑
- PointStamped ☑
- Path ☑
- Path ☑

Load 3D map (1, 2, 3), 4 is to add the rviz debugging tool.



- Global Options
- Global Status: Ok
- Grid ☑
- TF □
- LaserScan ☑
- Bumper Hit ☑
- Map ☑
- Global Map ☑
- Local Map ☑
- Amcl Particles ☑
- Rtabmap cloud ☑
  - Status: Ok
  - Topic /rtabmap/mapData
  - Unreliable □
  - Style Points
  - Size (Pixels) 3
  - Alpha 1
  - Position Transformer XYZ
  - Color Transformer RGB8
  - Cloud from scan □
  - Cloud decimation 4
  - Cloud max depth (m) 4
  - Cloud min depth (m) 0
  - Cloud voxel size (m) 0.05
  - Filter floor (m) 0
  - Filter ceiling (m) 0
  - Node filtering radius (m) 0
  - Node filtering angle (degr... 0
  - Download map ☑
  - Download graph □
- RobotModel ☑

Download map
Download the optimized global map using rtabmap/GetMap service.
This will force to re-create all clouds.

Calling "/rtabmap/get_map_data" service...

Downloading the map... please wait (rviz could become gray!)

OK

At this time, you can manually add [MarkerArray] to facilitate multi-point navigation and observation, and adjust [rviz] display parameters according to needs, such as the size of lidar points.

3.1 Single point navigation

- Use the [2D Pose Estimate] of the [rviz] tool to set the initial pose until the position of the car in the simulation is consistent with the actual position of the car.
- Click [2D Nav Goal] of the [rviz] tool, and then select a target point on the map where there are no obstacles. Release the mouse to start navigation. You can only select one target point and stop when you reach it.

3.2 Multi-point navigation

- Same as the first step of single-point navigation, first set the initial pose of the car.
- Click [Publish Point] of the [rviz] tool, and then select the target point on the map where there are no obstacles. Release the mouse to start navigation. You can click [Publish Point] again, and then select the point. The robot Will cruise between points.
- When using the [2D Pose Estimate] tool of the [rviz] tool to set the initial pose of the car, the multi-point navigation function is automatically canceled.

4. rtabmap node description

This is the master node for this package. It is a wrapper around the RTAB mapping core library. Here, when loop closure is detected, the map is incrementally built and optimized.

The node's online output is this map, which contains the latest data added to the map. The default location of the RTAB map database is [.ros/rtabmap.db], and the workspace is also set to [.ros].

4.1 Subscribed Topics

Name	Type	Analyze
odom	nav_msgs/Odometry	Odometry stream. Required if parameters `subscribe_depth` or `subscribe_stereo` are true **and** `odom_frame_id` is not set.
rgb/image	sensor_msgs/Image	RGB/Mono image
rgb/camera_info	sensor_msgs/CameraInfo	RGB camera metadata.
depth/image	sensor_msgs/Image	Registered depth image.
scan	sensor_msgs/LaserScan	Laser scan stream.
scan_cloud	sensor_msgs/PointCloud2	Laser scan point cloud stream.
left/image_rect	sensor_msgs/Image	Left RGB/Mono rectified image.
left/camera_info	sensor_msgs/CameraInfo	Left camera metadata
right/image_rect	sensor_msgs/Image	Right Mono rectified image.

Name	Type	Analyze
right/camera_info	sensor_msgs/CameraInfo	Right camera metadata
goal	geometry_msgs/PoseStamped	Plan a path to reach this goal using the current online map.
rgbd_image	rtabmap_ros/RGBDImage	RGB-D synchronized image, only when `subscribe_rgbd` is `true`.

4.2 Published Topics

Name	Type	Analyze
info	rtabmap_ros/Info	RTAB-Map's info.
mapData	rtabmap_ros/MapData	RTAB-Map's graph and latest node data.
mapGraph	rtabmap_ros/MapGraph	RTAB-Map's graph only.
grid_map	nav_msgs/OccupancyGrid	Occupancy grid generated with laser scans
proj_map	nav_msgs/OccupancyGrid	DEPRECATED: use `/grid_map` topic instead with `Grid/FromDepth=true`.
cloud_map	sensor_msgs/PointCloud2	3D point cloud generated from the assembled local grids
cloud_obstacles	sensor_msgs/PointCloud2	3D point cloud of obstacles generated from the assembled local grids
cloud_ground	sensor_msgs/PointCloud2	3D point cloud of ground generated from the assembled local grids.
scan_map	sensor_msgs/PointCloud2	3D point cloud generated with the 2D scans or 3D scans
labels	visualization_msgs/MarkerArray	Convenient way to show graph's labels in RVIZ.
global_path	nav_msgs/Path	Poses of the planned global path. Published only once for each path planned.
local_path	nav_msgs/Path	Upcoming local poses corresponding to those of the global path. Published on every map update.
goal_reached	std_msgs/Bool	Status message if the goal is successfully reached or not.
goal_out	geometry_msgs/PoseStamped	Current metric goal sent from the rtabmap's topological planner. For example, this can be connected move_base_simple/goal to move_base。
octomap_full	octomap_msgs/Octomap	Get an OctoMap. Available only if rtabmap_ros is built with octomap.

Name	Type	Analyze
octomap_binary	octomap_msgs/Octomap	Get an OctoMap. Available only if rtabmap_ros is built with octomap.
octomap_occupied_space	sensor_msgs/PointCloud2	A point cloud of the occupied space (obstacles and ground) of the OctoMap. Available only if rtabmap_ros is built with octomap.
octomap_obstacles	sensor_msgs/PointCloud2	A point cloud of the obstacles of the OctoMap. Available only if rtabmap_ros is built with octomap.
octomap_ground	sensor_msgs/PointCloud2	A point cloud of the ground of the OctoMap. Available only if rtabmap_ros is built with octomap.
octomap_empty_space	sensor_msgs/PointCloud2	A point cloud of empty space of the OctoMap. Available only if rtabmap_ros is built with octomap.
octomap_grid	nav_msgs/OccupancyGrid	The projection of the OctoMap into a 2D occupancy grid map. Available only if rtabmap_ros is built with octomap.

4.3 Services

Name	Type	Analyze
get_map	rtabmap_ros/GetMap	Call this service to get the standard 2D occupancy grid
get_map_data	rtabmap_ros/GetMap	Call this service to get the map data
publish_map	rtabmap_ros/PublishMap	Call this service to publish the map data
list_labels	rtabmap_ros/ListLabels	Get current labels of the graph.
update_parameters	std_srvs/Empty	The node will update with current parameters of the rosparam server
reset	std_srvs/Empty	Delete the map
pause	std_srvs/Empty	Pause mapping
resume	std_srvs/Empty	Resume mapping
trigger_new_map	std_srvs/Empty	The node will begin a new map
backup	std_srvs/Empty	Backup the database to "database_path.back" (default ~/.ros/rtabmap.db.back)
set_mode_localization	std_srvs/Empty	Set localization mode
set_mode_mapping	std_srvs/Empty	Set mapping mode
set_label	rtabmap_ros/SetLabel	Set a label to latest node or a specified node.

Name	Type	Analyze
set_goal	rtabmap_ros/SetGoal	Set a topological goal (a node id or a node label in the graph).
octomap_full	octomap_msgs/GetOctomap	Get an OctoMap. Available only if rtabmap_ros is built with octomap.
octomap_binary	octomap_msgs/GetOctomap	Get an OctoMap. Available only if rtabmap_ros is built with octomap.

4.4 Parameters

Name	Type	Default value	Analyze
subscribe_depth	bool	true	Subscribe to depth image
subscribe_scan	bool	false	Subscribe to laser scan
subscribe_scan_cloud	bool	false	Subscribe to laser scan point cloud
subscribe_stereo	bool	false	Subscribe to stereo images
subscribe_rgbd	bool	false	Subsribe to rgbd_image topic
frame_id	string	base_link	The frame attached to the mobile base.
map_frame_id	string	map	The frame attached to the map.
odom_frame_id	string	‘ ’	The frame attached to odometry.
odom_tf_linear_variance	double	0.001	When odom_frame_id is used, the first 3 values of the diagonal of the 6x6 covariance matrix are set to this value.
odom_tf_angular_variance	double	0.001	When odom_frame_id is used, the last 3 values of the diagonal of the 6x6 covariance matrix are set to this value.
queue_size	int	10	Size of message queue for each synchronized topic.
publish_tf	bool	true	Publish TF from /map to /odom.
tf_delay	double	0.05	
tf_prefix	string	‘ ’	Prefix to add to generated tf.
wait_for_transform	bool	true	Wait (maximum wait_for_transform_duration sec) for transform when a tf transform is not still available.
wait_for_transform_duration	double	0.1	Wait duration for wait_for_transform.
config_path	string	‘ ’	Path of a config files containing RTAB-Map's parameters. Parameters set in the launch file overwrite those in the config file.
database_path	string	.ros/rtabmap.db	Path of the RTAB-Map's database.

Name	Type	Default value	Analyze
gen_scan	bool	false	Generate laser scans from depth images (using the middle horizontal line of the depth image). Not generated if subscribe_scan or subscribe_scan_cloud are true.
gen_scan_max_depth	double	4.0	Maximum depth of the laser scans generated.
approx_sync	bool	false	Use approximate time synchronization of input messages. If false, note that the odometry input must have also exactly the same timestamps than the input images.
rgbd_cameras	int	1	Number of RGB-D cameras to use (when subscribe_rgbd is true). Well for now, a maximum of 4 cameras can be synchronized at the same time.
use_action_for_goal	bool	false	Use actionlib to send the metric goals to move_base.
odom_sensor_sync	bool	false	Adjust image and scan poses relatively to odometry pose for each node added to graph.
gen_depth	bool	false	Generate depth image from scan_cloud projection into RGB camera, taking into account displacement of the RGB camera accordingly to odometry and lidar frames.
gen_depth_decimation	int	1	Scale down image size of the camera info received (creating smaller depth image).
gen_depth_fill_holes_size	int	0	Fill holes of empty pixels up to this size. Values are interpolated from neighbor depth values. 0 means disabled.
gen_depth_fill_iterations	double	0.1	Maximum depth error (m) to interpolate.

Name	Type	Default value	Analyze
gen_depth_fill_holes_error	int	1	Number of iterations to fill holes.
map_filter_radius	double	0.0	Filter nodes before creating the maps. Only load data for one node in the filter radius (the latest data is used) up to filter angle (map_filter_angle).
map_filter_angle	double	30.0	Angle used when filtering nodes before creating the maps. See also map_filter_radius. Used for all published maps.
map_cleanup	bool	true	If there is no subscription to any map cloud_map, grid_map or proj_map, clear the corresponding data.
latch	bool	true	If true, the last message published on the map topics will be saved and sent to new subscribers when they connect.
map_always_update	bool	true	Always update the occupancy grid map even if the graph has not been updated
map_empty_ray_tracing	bool	true	Do ray tracing to fill unknown space for invalid 2D scan's rays (assuming invalid rays to be infinite). Used only when map_always_update is also true.

4.5 TF transform

- Preparation:
  - base_link → sensor
  - odom → base_link

  Provide:

  map → odom