

```
roslaunch transbot_nav transbot_bringup.launch # Astra + laser + Transbot
roslaunch transbot_nav astra_bringup.launch    # Astra + Transbot
roslaunch transbot_nav transbot_rtabmap.launch open_rviz:=False # mapping
roslaunch transbot_nav transbot_rtabmap_nav.launch open_rviz:=False # navigation
```

Note: Astra + laser + Transbot is the fusion of depth camera and lidar; Astra + Transbot refers to pure vision, mainly using the function package `depthimage_to_laserscan` to convert the depth image into lidar data (the scanning range is different from lidar), and its mapping function Same as lidar.

Start visualization

```
roslaunch transbot_nav view_rtabmap.launch      # mapping visualization
roslaunch transbot_nav view_rtabmap_nav.launch  # navigation visualization
```

View tf tree

```
roslaunch rqt_tf_tree rqt_tf_tree
```

View node

```
rqt_graph
```

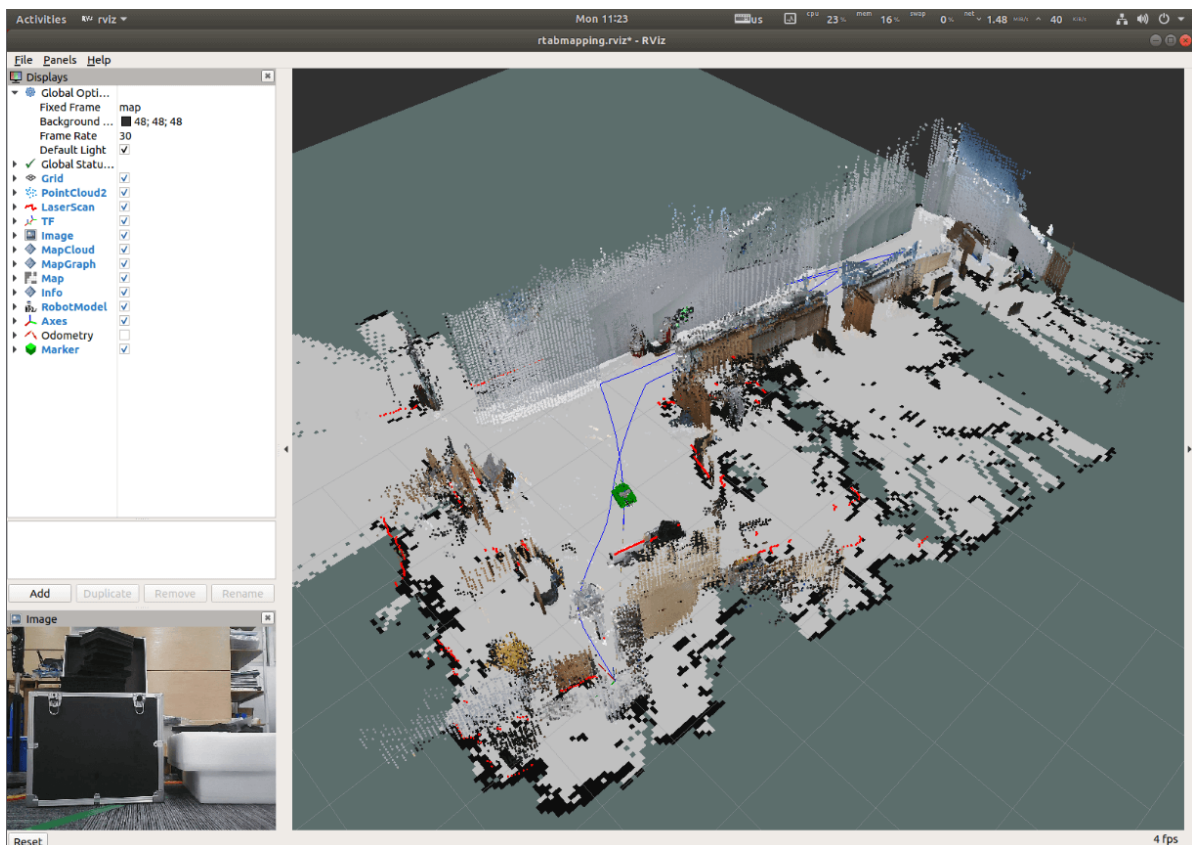
Keyboard control node

```
roslaunch teleop_twist_keyboard teleop_twist_keyboard.py # system integration
roslaunch transbot_ctrl transbot_keyboard.launch         # customize
```

9.2、 Navigation

After starting up according to the above method, you can choose any method to control the map creation (handle control is recommended); the slower the map creation, the better the effect (especially the angular speed); the robot is full of the area to be created.

- Handle control
- Keyboard control



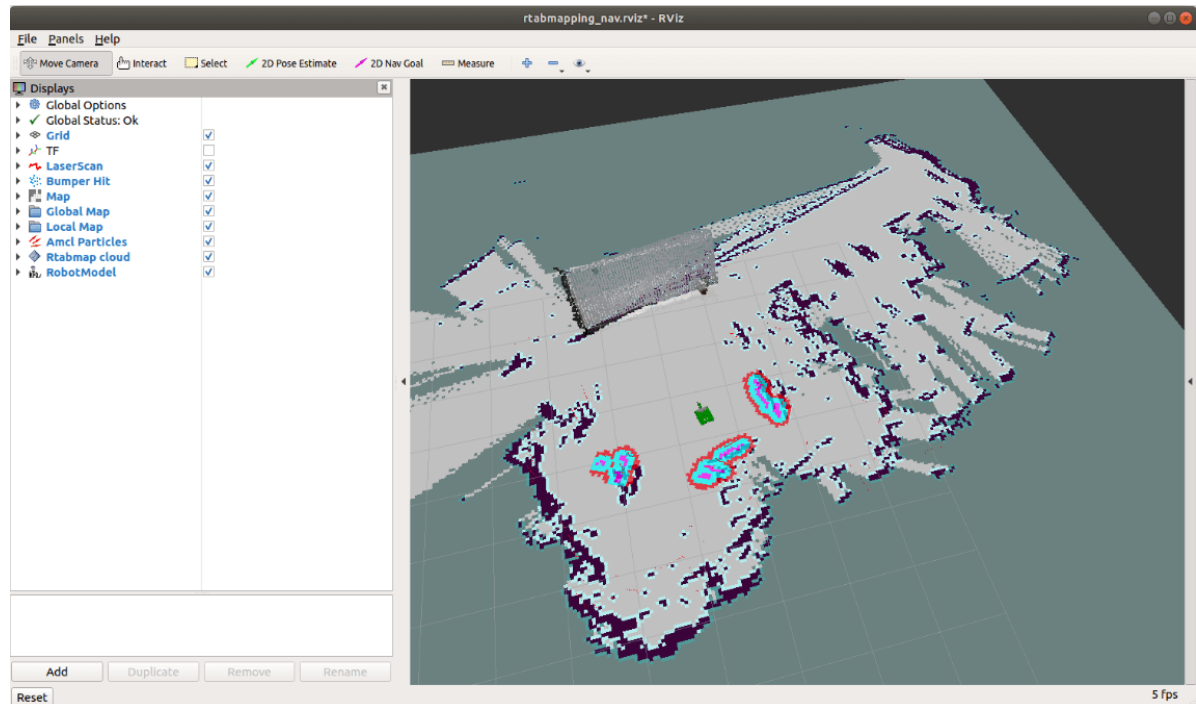
When the map is completed, directly ctrl+c to exit the map node, the system will automatically save the map.

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

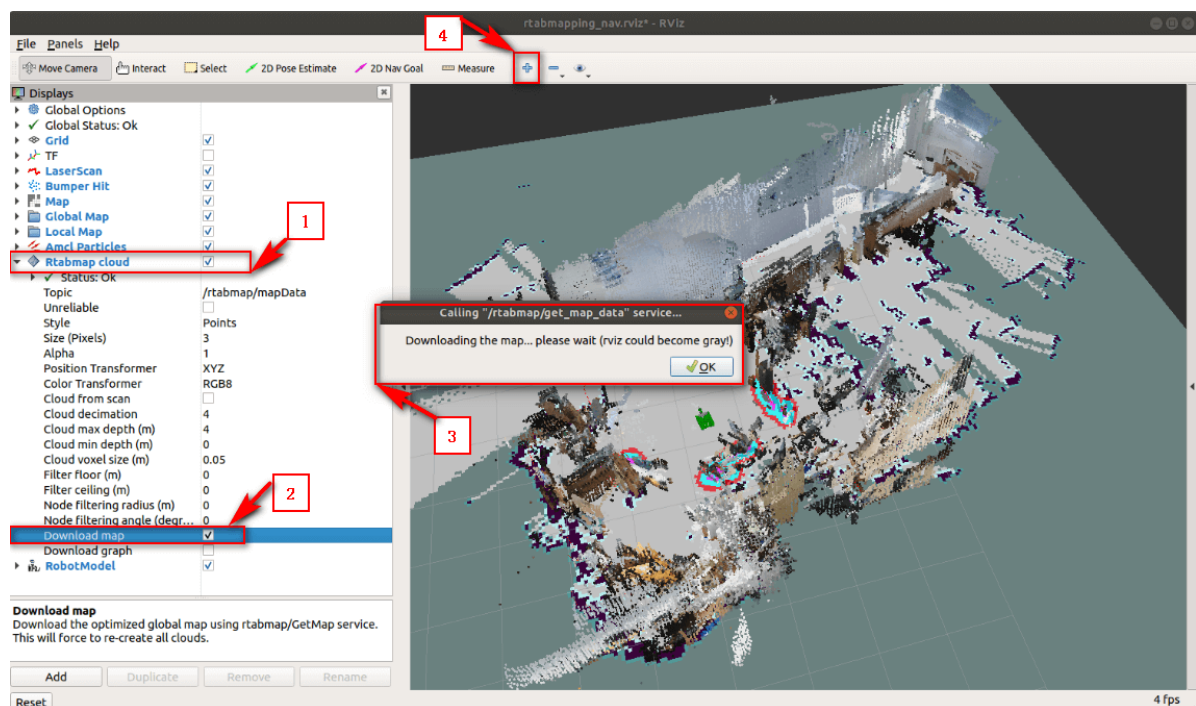
9.3、Navigation obstacle avoidance

Note: [R2] on the handle can cancel the target point.

When the navigation mode is turned on, the system automatically loads a 2D raster map, and cannot directly load a 3D map. It needs to be loaded manually.



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



Steps for usage

- Place the robot at the origin and start the navigation function, the system will automatically match the position of the robot.

- Single-point navigation:

Click the **【2D Nav Goal】** of the **【rviz】** tool. Then use the mouse to select a target point on the map model where there are no obstacles. Release the mouse to start the navigation. Only one target point can be selected. Finally, robot car will move towards the target point form, after reaching the target point, the car will stop.

- Multi-point navigation:

Click **【Publish Point】** of the **【rviz】** tool. Then select a target point where there are no obstacles on the map model, release the mouse to start navigation. You can click **【Publish Point】** again, then select others point. Finally, robot car will cruise from point to point.

9.4、Node rtabmap

9.4.1、Subscribed Topics

Name	Type	Analyze
odom	nav_msgs/Odometry	Odometry stream. Required if parameters <code>subscribe_depth</code> or <code>subscribe_stereo</code> are true and <code>odom_frame_id</code> 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.
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 <code>subscribe_rgbd</code> is true.

9.4.2、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 <code>/grid_map</code> topic instead with <code>Grid/FromDepth=true</code> .
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 <code>move_base_simple/goal</code> to <code>move_base</code> .
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.

9.4.3、 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.
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.

9.4.4、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	Subscribe 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.
gen_depth_fill_holes_error	int	1	Number of iterations to fill holes.

Name	Type	Default value	Analyze
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.

9.4.5、tf transform

Preparation:

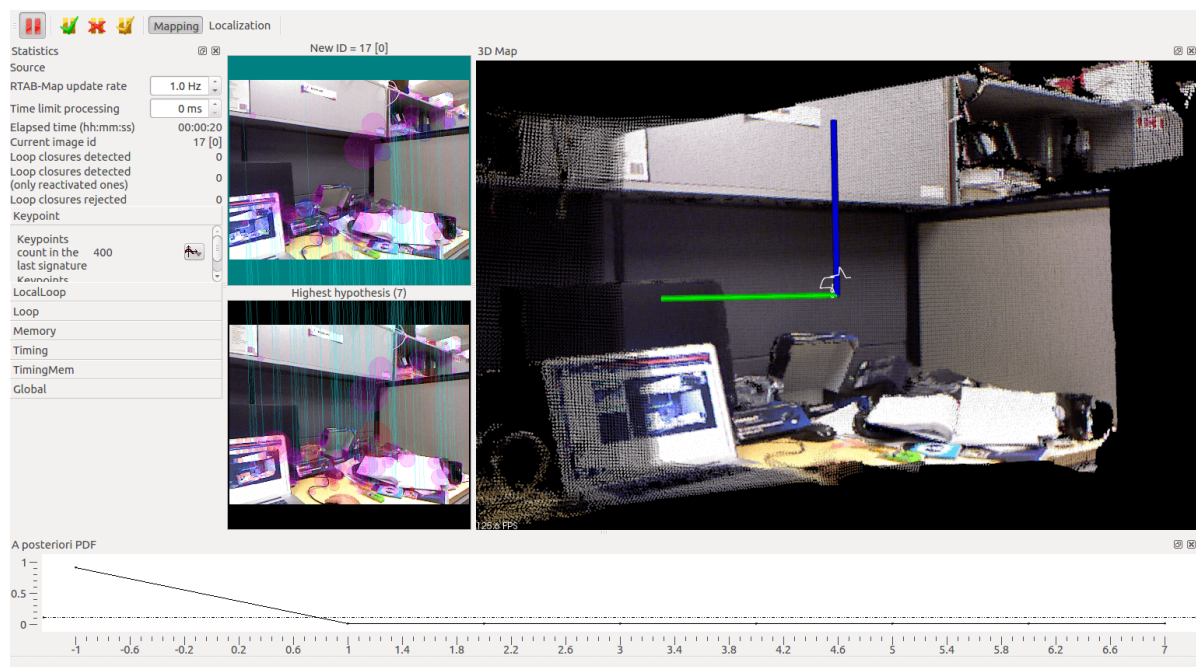
- base_link → sensor
- odom → base_link

Provide:

map → odom

9.5、Node rtabmapviz

This node starts the visualization interface of RTAB-Map. It is a wrapper for the RTAB-MapGUI library. Its purpose is the same as rviz, but with RTAB-Map specific options.



9.5.1、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. Should be rectified when subscribe_depth is true. Not required if parameter subscribe_stereo is true (use left/image_rect instead).
rgb/camera_info	sensor_msgs/CameraInfo	RGB camera metadata. Not required if parameter subscribe_stereo is true (use left/camera_info instead).
depth/image	sensor_msgs/Image	Registered depth image. Required if parameter subscribe_depth is true.
scan	sensor_msgs/LaserScan	Laser scan stream. Required if parameter subscribe_scan is true.
scan_cloud	sensor_msgs/PointCloud2	Laser scan stream. Required if parameter subscribe_scan_cloud is true.
left/image_rect	sensor_msgs/Image	Left RGB/Mono rectified image. Required if parameter subscribe_stereo is true.
left/camera_info	sensor_msgs/CameraInfo	Left camera metadata. Required if parameter subscribe_stereo is true.
right/image_rect	sensor_msgs/Image	Right Mono rectified image. Required if parameter subscribe_stereo is true.
right/camera_info	sensor_msgs/CameraInfo	Right camera metadata. Required if parameter subscribe_stereo is true.
odom_info	rtabmap_ros/OdomInfo	Odometry info. Required if parameter subscribe_odom_info is true.
info	rtabmap_ros/Info	RTAB-Map's statistics info.
mapData	rtabmap_ros/MapData	RTAB-Map's graph and latest node data.
rgbd_image	rtabmap_ros/RGBDImage	RGB-D synchronized image, only when subscribe_rgbd is true.

9.5.2、Parameters

Parameters	Type	Default value	Analyze
subscribe_depth	bool	false	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_odom_info	bool	false	Subscribe to odometry info messages
subscribe_rgbd	bool	false	Subscribe to <code>rgb_d_image</code> topic.
frame_id	string	base_link	The frame attached to the mobile base.
odom_frame_id	string	''	The frame attached to odometry. If empty, rtabmapviz will subscribe to odom topic to get odometry. If set, odometry is got from tf.
tf_prefix	string	''	Prefix to add to generated tf.
wait_for_transform	bool	false	Wait (maximum 1 sec) for transform when a tf transform is not still available.
queue_size	int	10	Size of message queue for each synchronized topic.
rgb_d_cameras	int	1	Number of RGB-D cameras to use (when subscribe_rgb_d is true). Well for now, a maximum of 4 cameras can be synchronized at the same time.

9.5.3、Required tf Transforms

- base_link → Sensor coordinate system
- odom → base_link
- map → odom