version 1.1

This document is published for international developers who have no access to the academy.
You are encouraged to update this document for the latest version of Autoware.

TIER IV ACADEMY
BUILDING AUTONOMOUS DRIVING SYSTEM

Autoware Hands-on Exercises

Tier IV
Intelligent Vehicle

Table of Contents

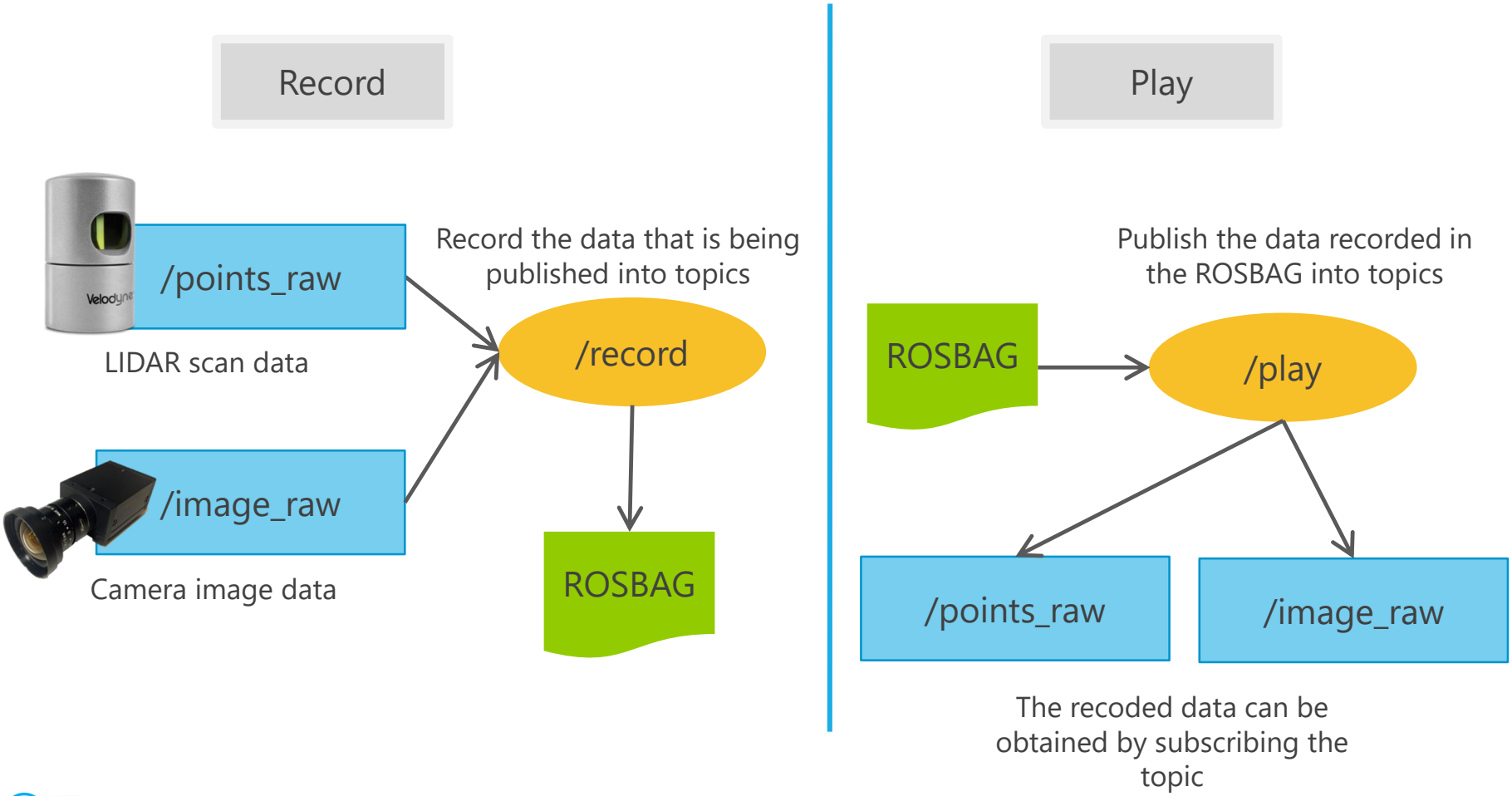
Autoware Hands on Exercises

Chapter 1：Data Recording/Playing and Sensor Calibration

1. Data Recording/Playing

Data Recording/Playing – Overview

- In ROS, log data of sensor and etc., can be saved into a file format called ROSBAG
- Playing ROSBAG replays recorded sensor information, it can be used for simulation
- Autoware provides ROSBAG recording/playing functions in GUI



Data Recording – Steps (1/2)

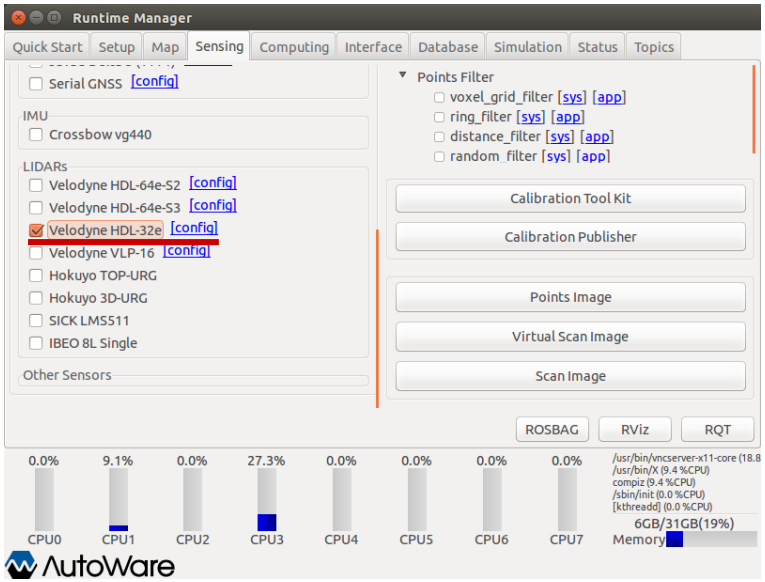
- Publish topics to be recorded

```
ando@ando-Diginnos-PC:~
$ cd ~/Autoware/ros/
ando@ando-Diginnos-PC:~/Autoware/ros
$ ./run
```

1. Launch Autoware
   A) Execute “run” on a terminal as follows (alternatively, click on the “run” script on a file manager) :

   $ cd ~/Autoware/ros/

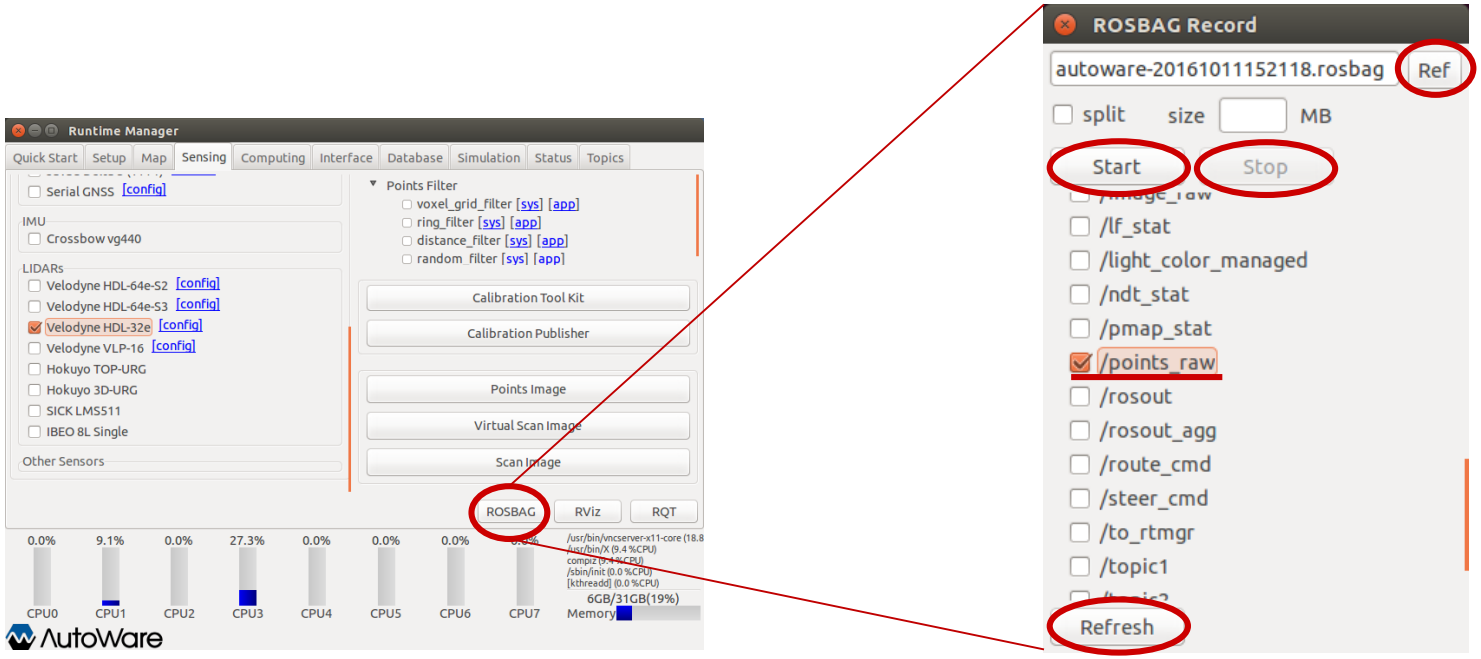
   $ ./run



2. Launch ROS nodes by checking (☑) the boxes, this will publish ROS topics that can be recorded
   - i.e. [Velodyne HDL-32e]: this node publishes Velodyne data to /points_raw topic

Data Recording – Steps (2/2)

- ROSBAG recording
  - A) Display a dialog by clicking [ROSBAG] button
  - B) Click [Ref] button, and then specify the filename of the ROSBAG to be saved
  - C) Click [Refresh] button
  - D) Check (☑) the boxes of ROS topics to be recorded
  - E) Click [Start] button to start recording the selected topic
  - F) Click [Stop] button to end the recording



Data Playing – Step

- ROSBAG playing

1. Playing a bag file in [Simulation] tab

A) Open [Simulation] tab

B) Click [Ref] button, and then specify the bag file to play

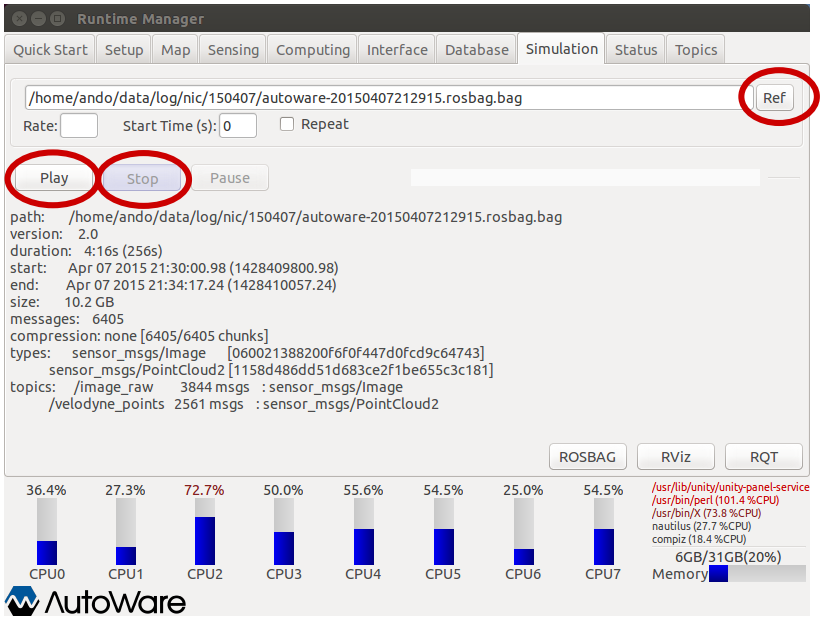
C) Click [Play] button to play back the contents of the bag file

- The ROS parameter [/use_sim_time] is set to [true] automatically

D) Click [Pause] button to pause/resume the playback

E) Click [Stop] button to stop the playback

- The ROS parameter [/use_sim_time] is set to [false] automatically



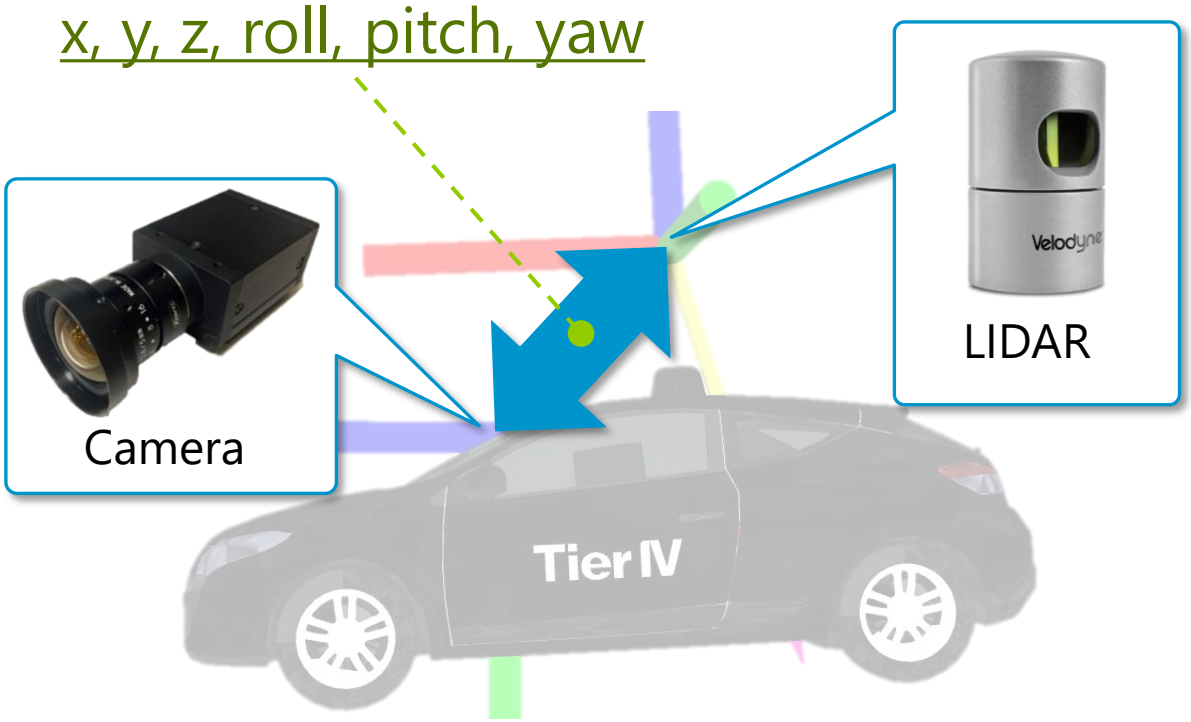
Autoware Hands-on Exercises

Chapter 1：Data Recording/Playing and Sensor Calibration

2. Sensor Calibration

Sensor Calibration – Overview

- Calculate relative position relation (i.e., x, y ,z, roll, pitch, yaw) of the LIDAR and the camera mounted on vehicle
- Both of the LIDAR data and the camera data can be fused by utilizing the relative position between them
  - The LIDAR scan data is projected on the camera image
- A checkerboard is used to calibrate both sensors



Sensor Calibration – Structure

/points_raw

LIDAR scan data

/image_raw

Camera image data

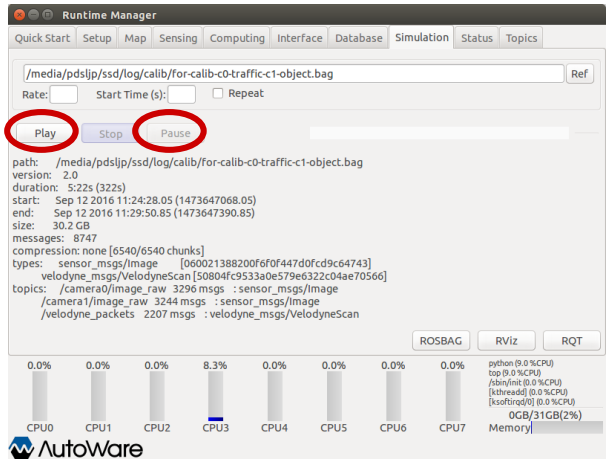
calibration_toolkit

Calibration.yaml

Calibration file

Sensor Calibration – Steps (1/4)

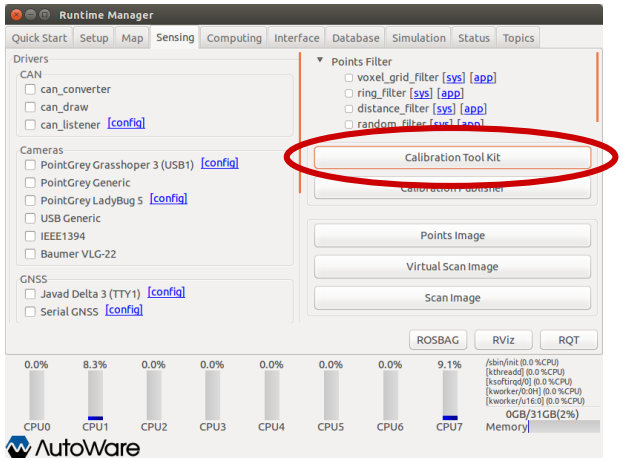
- Launch tools
  - In this slide, a calibration file is generated by using a ROSBAG that contains "/points_raw" and "/image_raw"



1. Play/Stop ROSBAG

A) Click [Play] button in [Simulation] tab, and then click [Pause] button

➢ If the ROSBAG includes "/velodyne_packets" instead of "/points_raw", launch Velodyne Driver following the steps described in p.18

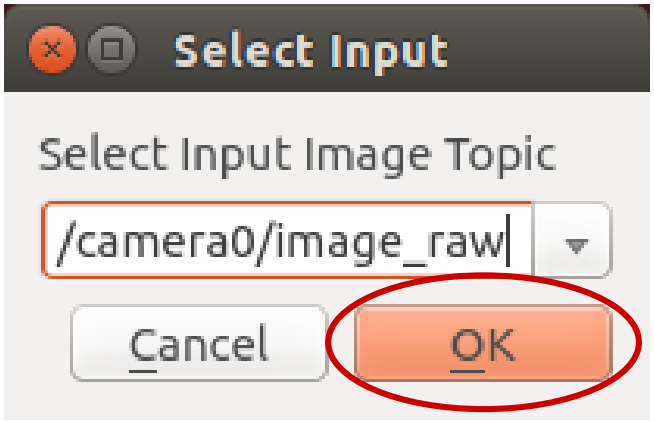


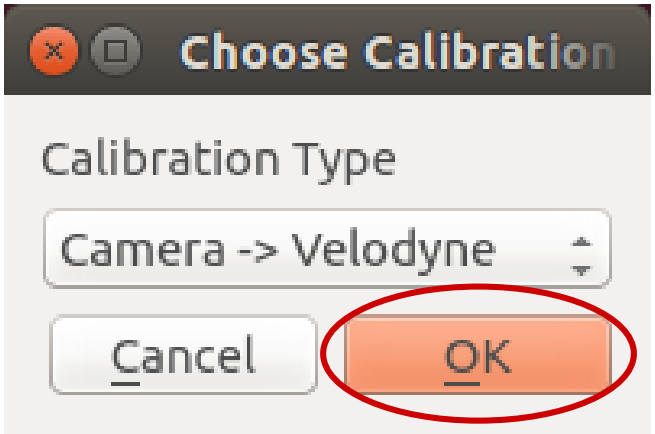
2. Launch Calibration Toolkit

A) Click [Calibration Toolkit] in [Sensing] tab

Sensor Calibration – Steps (2/4)

- Data source selection

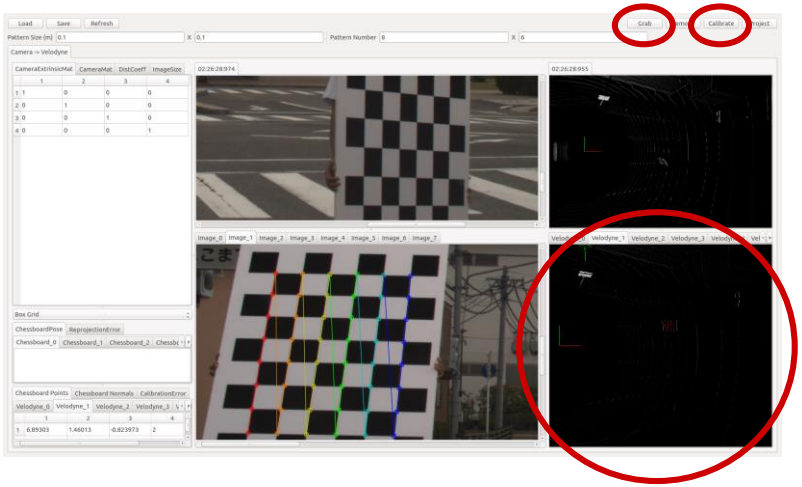




3. Select data source for calibration
   - A) Select a camera topic for calibration
     - Here, select "/camera0/image_raw"
   - B) Select calibration type
     - Since this example uses the relative position between the camera and the Velodyne, select [Camera->Velodyne]

Sensor Calibration – Steps (3/4)

- Run data grab and calibration



Move: ↑,↓,→,←,PgUp,PgDn
Rotate: a,d,w,s,q,e
Point size: o(decrease), p(increase)
Change background color: b

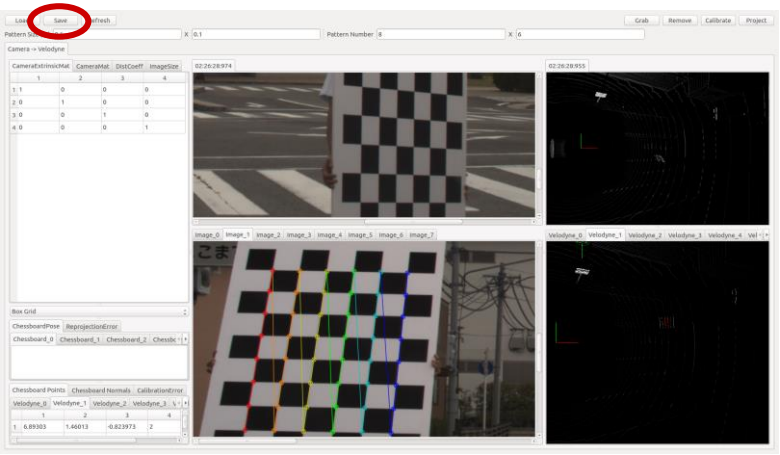
4. Press Grab to capture data from both sensors
   A) Confirm that a camera image (upper-left) and LIDAR (upper-right) data are displayed, then click [Grab] button

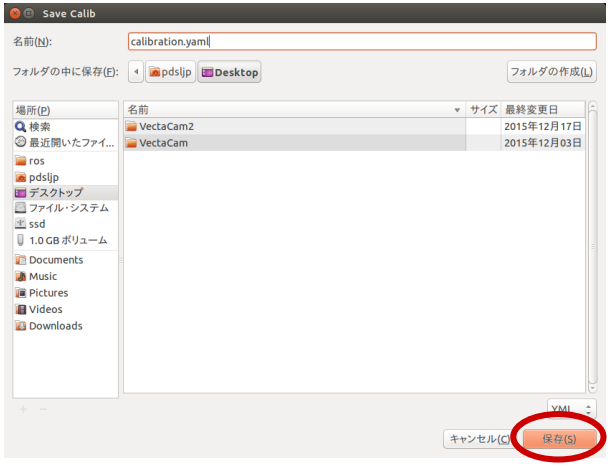
5. Select grabbed data points that are projected on the checkerboard
   A) Click LIDAR scan projected on the checkerboard from the right-bottom window
      - Conduct this step as many times as the number of grabbed frames.

6. Calibration
   A) Click [Calibrate]
   B) Confirm that the parameters of calibration are updated on the table shown in the left window

Sensor Calibration – Steps (4/4)

- Save calibration data file





7. Save file
   - A) Click [Save] button
   - B) Select the file name, the directory, and then click the [Save] button

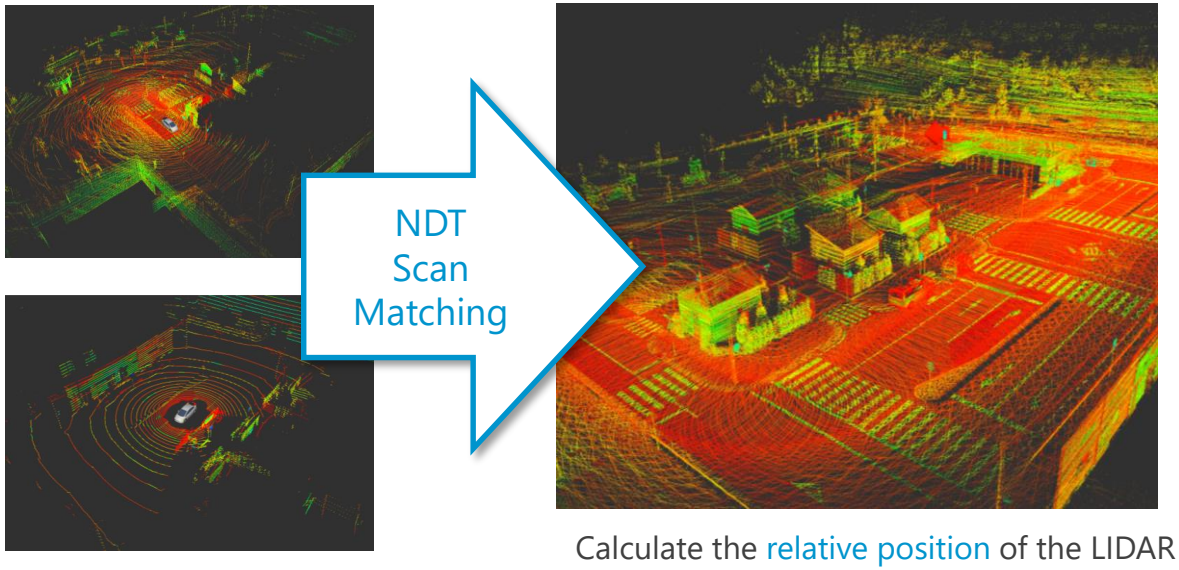
Autoware Hands-on Exercises

Chapter 2：3D Map Generation and Localization

1. 3D Map Generation

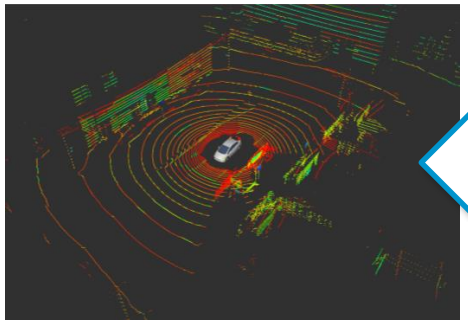
3D Map Generation – Overview

- Generate 3D map from LIDAR scan data
- Localize the vehicle position with the help of NDT scan matching, and then add the scan data to the initial 3D map
  - The larger the map data, the longer the calculation time required for localization. The 3D map is generated by recording and then playing LIDAR points on the initial 3D map
- 3D map is down-sampled with the help of Voxel Grid filter. Finally, the down sampled map is stored in a PCD file.
- If the size of map is large, the error in the 3D map may also be large.

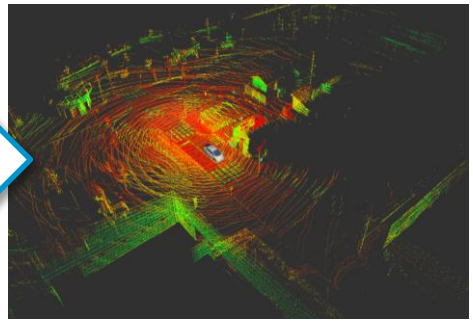


Calculate the relative position of the LIDAR from matching result of both scan data, and then add the new scan data to the 3D map

3D Map Generation – Structure



NDT
Scan Matching



/points_raw

LIDAR scan data

/ndt_mapping

/ndt_map

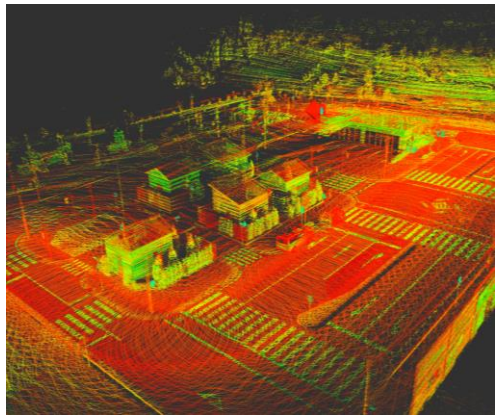
Generated 3D map

PCD

Down-sampling by
VoxelGrid Filter

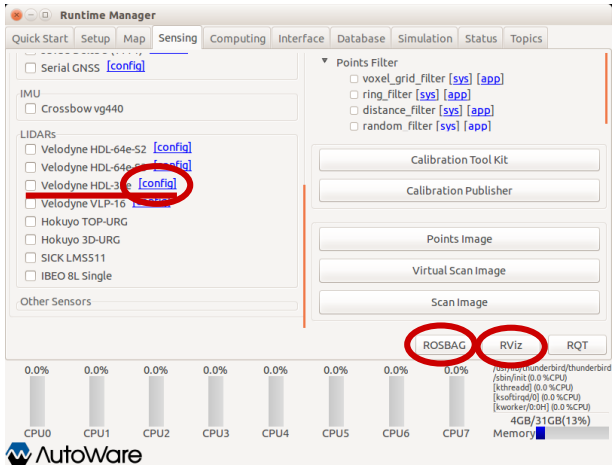
/current_pose

Localization



3D Map Generation - Steps (1/5)

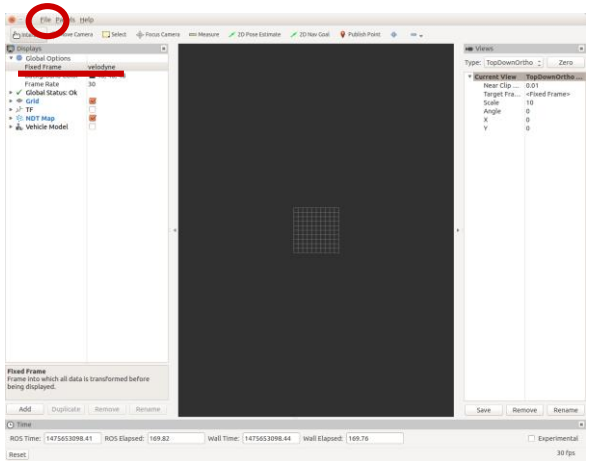
- ROSBAG Recording



1. Record LIDAR scan data into a ROSBAG

A) Launch Velodyne HDL-32e driver (a ROS node)

- Click [config] on [Velodyne HDL-32e] in [Sensing] tab, and specify the following file:
  Autoware/ros/src/sensing/drivers/lidar/packages/velodyne/velodyne_pointcloud/params/32db.yaml
- Check (☑) the box of [Velodyne HDL-32e]

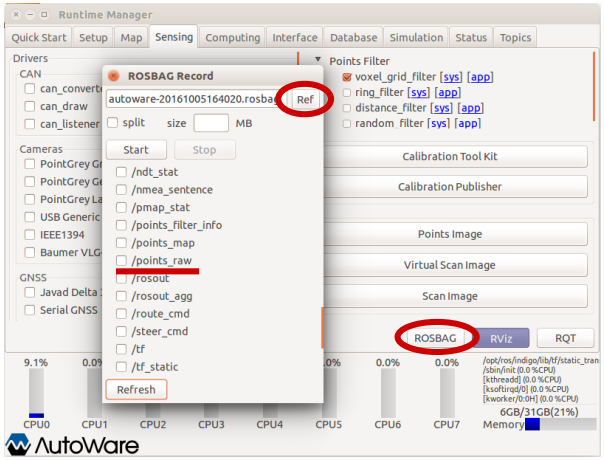


B) Confirm the ROSBAG data on RViz by clicking [Rviz] button

- Click [File] - [Open Config], and then open the following file:
  Autoware/ros/src/.config/rviz/ndt_mapping.rviz
- Change [Fixed Frame] on [Global Options] in RViz to [velodyne]
- Confirm that [Points Raw] is being displayed

3D Map Generation - Steps (2/5)

- ROSBAG logging

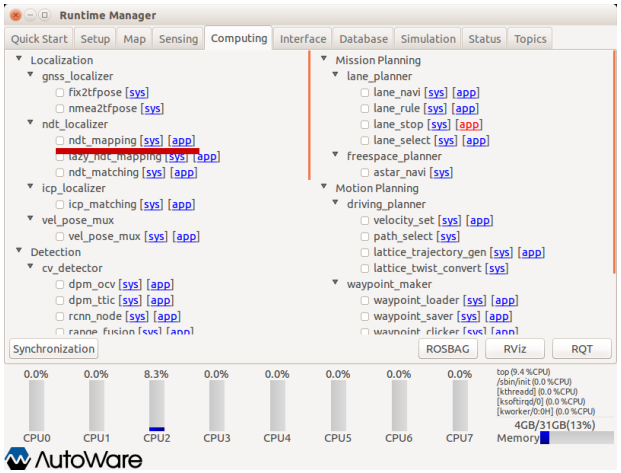


C) Click [ROSBAG] and record ROSBAG

- Click [Ref] button and specify the filename and directory of the ROSBAG to be saved.
- Check (☑) the box of ROS topic [/points_raw]
- Click [Start] to start the ROSBAG recording
- Click [Stop] to end the ROSBAG recording

3D Map Generation - Steps (3/5)

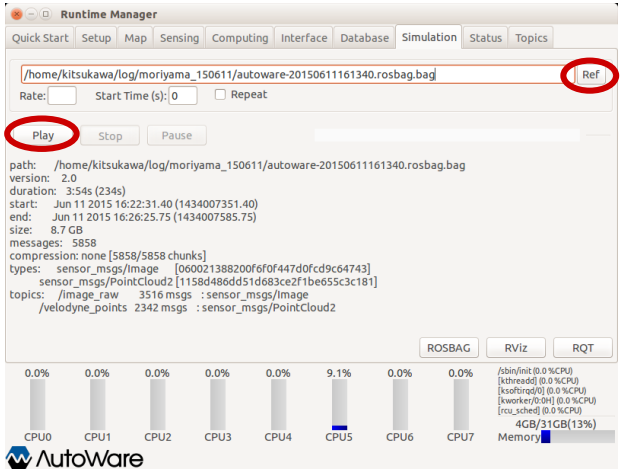
- 3D map generation



2. 3D map generation while playing LIDAR points

A) Launch the map generation node [ndt_mapping]

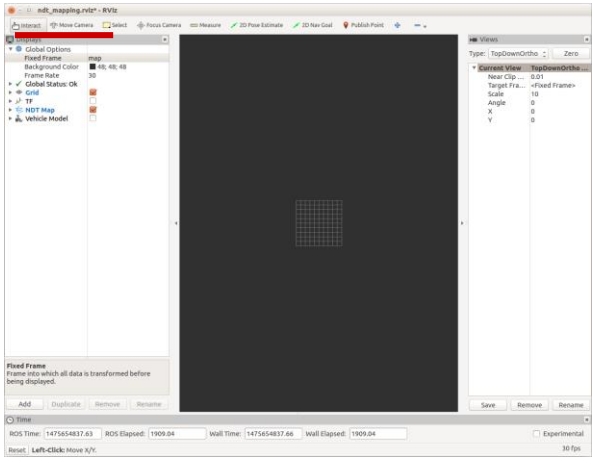
- Launch [ndt_mapping] by checking (☑) the box in [Computing] tab



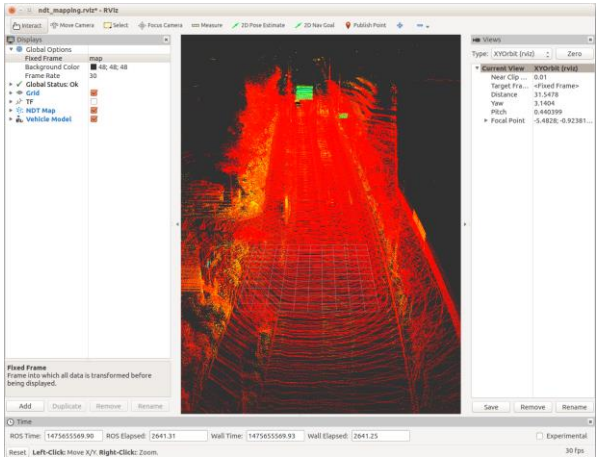
B) ROSBAG playback

- Click [Ref] on [Simulation] tab, and specify the ROSBAG recorded scan data
- Click [Play] to playback

3D Map Generation - Steps (4/5)



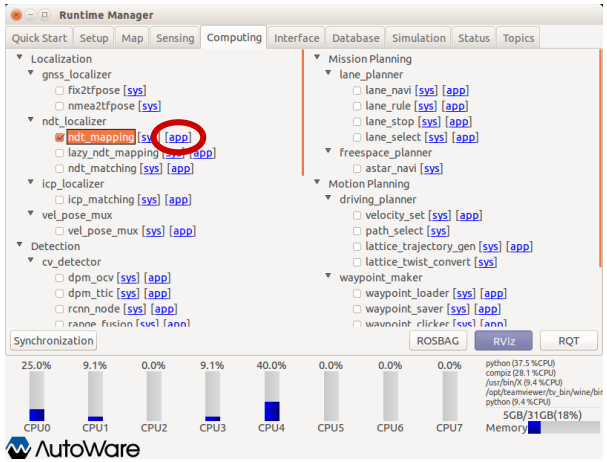
C) Change [Fixed Frame] on [Global Options] in RViz to [map]



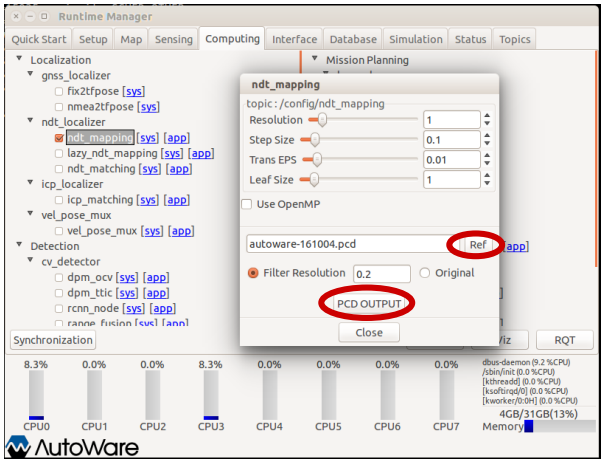
D) With the help of RViz confirm that the 3D map is being generated

3D Map Generation - Steps (5/5)

- Save 3D map



3. Save 3D map in a PCD file after generating the 3D map
   A) Open [Config] window by clicking [app] on [ndt_mapping] in [Computing] tab



   B) Click [Ref], specify the directory and the filename to save the map into a PCD file
   C) Specify sampling parameter [Filter Resolution] (default: 0.2) for 3D map, and click [PCD OUTPUT] button to start saving
   D) Confirm the file exists in the specified directory

Autoware Hands-on Exercises

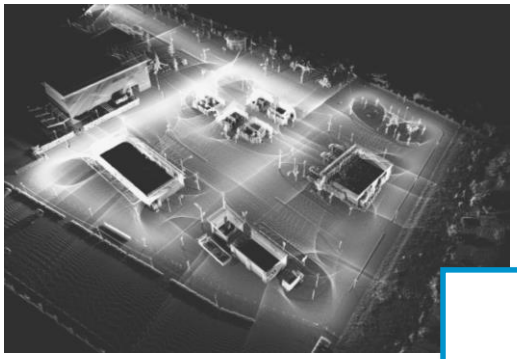
Chapter 2：3D Map generation and Localization

2. Localization

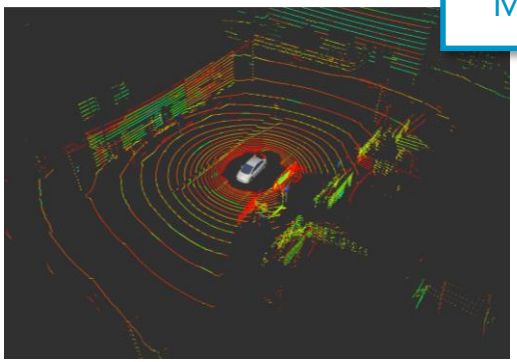
Localization – Overview

- Calculate self-position (the position and the direction in the map) by scan matching with LIDAR scan data and a 3D map
- Specify the initial localization position either using the RViz button [2D Pose Estimate], GNSS data or known map coordinate values
- Voxel Grid filter node is required by localization node

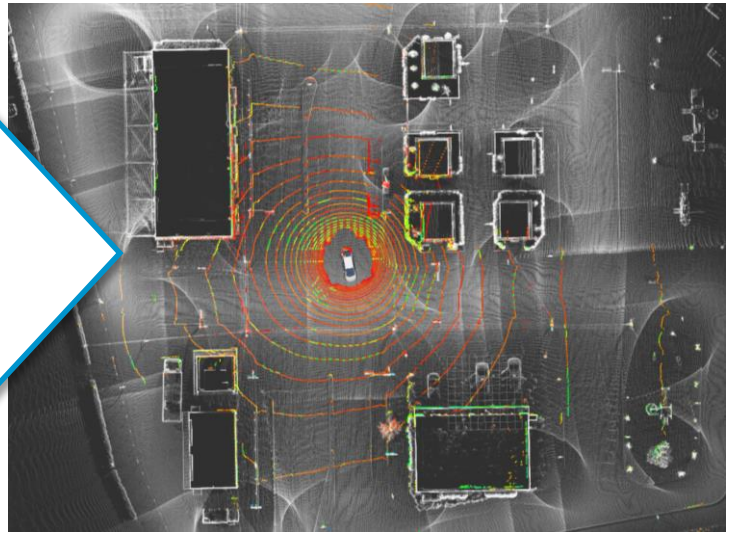
Map data



Scan data



NDT Scan Matching



Estimate the position and moving direction of the vehicle from the best matching information between the map data and the scan data

Localization – Workflow

points_raw

voxel_grid_filter

Reduce points for matching by sampling scan data

filtered_ points

points_map_loader

Load map data (PCD) and publish it

PCD

points_map

ndt_matching

Estimate position by NDT scan matching using the 3D map and the scan data

ndt_pose

current_pose
position (x,y,z) ・direction (roll,pitch,yaw)

initialpose (RViz)

Specify the initial position for localization

gnss_pose

vel_pose_mux

Decide the position and the direction for vehicle control

current_pose

Vehicle control

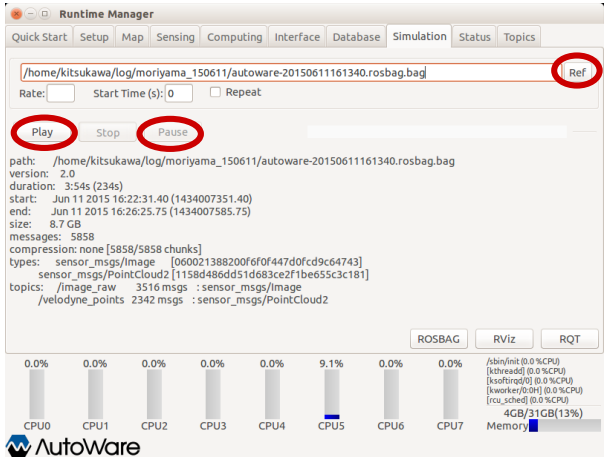
nmea_ sentence

nmea2tfpose

Convert the position information of NMEA format into xyz format

Localization – Steps (1/5)

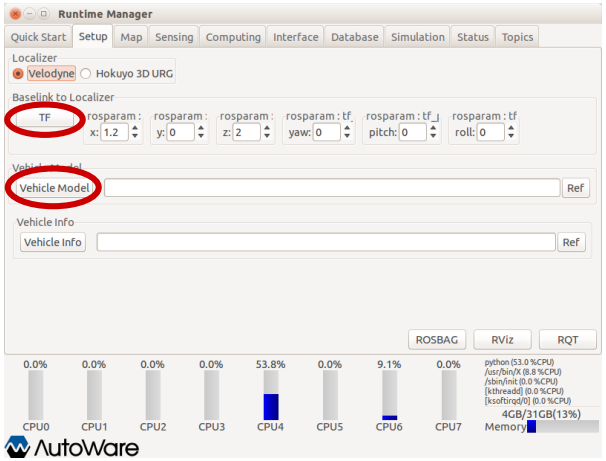
- Specifying simulation time, TF, and vehicle model



1. Turning Simulation Time "ON"

A) Select a ROSBAG file in [Simulation] tab for localization. Click [Play], and then click [Pause].

- This operation automatically sets use_sim_time of ROS Param to "ON"



2. TF set and vehicle model loading

A) Select [Velodyne] on [Localizer] in [Setup] tab

B) Specify the relative position between the vehicle control position and the Velodyne on [Baselink to Localizer] as follows:

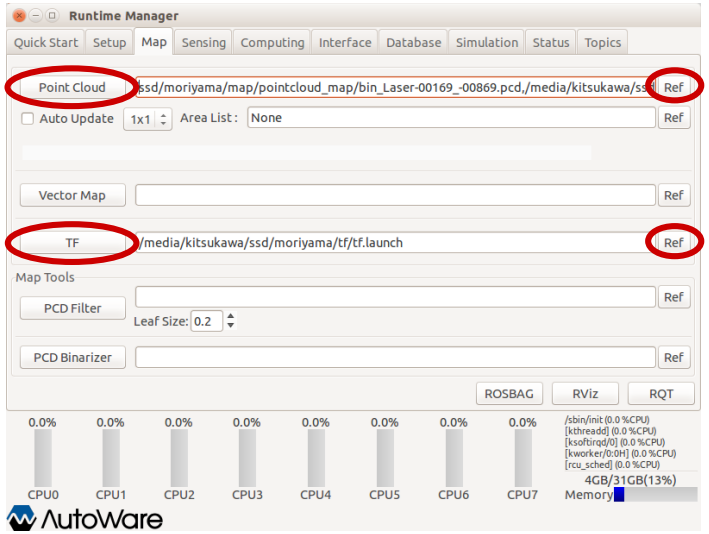
Specify x:1.2, y:0, z:2.0, yaw:0, pitch:0, roll:0, and then click [TF]

C) Click [Vehicle Model]

- If no file is specified, default model is loaded

Localization – Steps (2/5)

- Map data loading

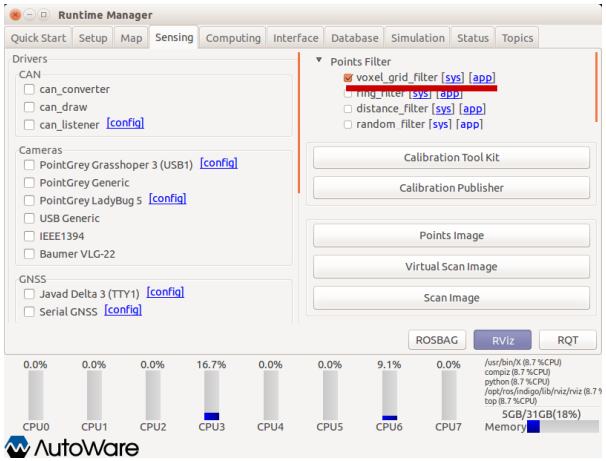


3. Point cloud map loading
   - A) Click [Ref] of [Point Cloud] in [Map] tab, specify the PCD file generated by map generation
   - B) Click [Point Cloud] button

4. Specify the TF representing the position of the map data
   - A) Click [Ref] on [TF] in [Map] tab and specify "tf_local.launch"
   - B) Click [TF] button
     - ➢ This operation publishes TF between world frame and map frame

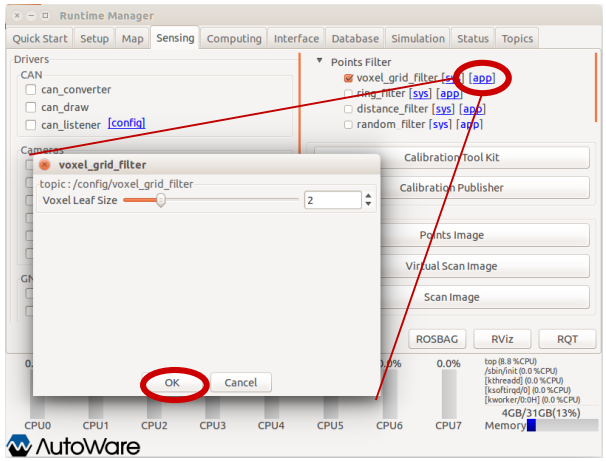
Localization – Steps (3/5)

- Map data Loading



5. Down-sampling scan data

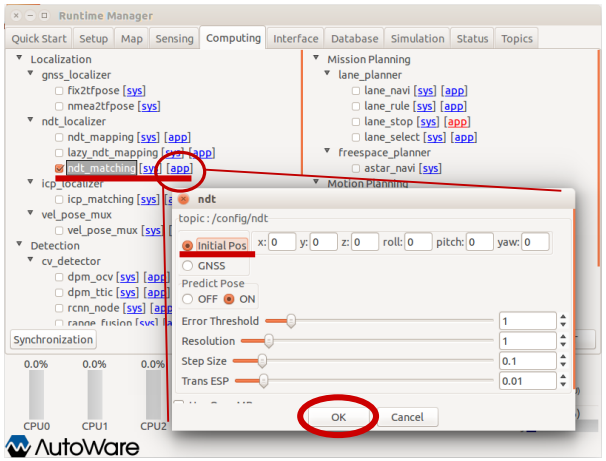
   A) Launch [voxel_grid_filter] in [Sensing] tab by checking (☑) the box



   B) Click [app], specify [Voxel Leaf Size] (default: 2.0) in the shown window, and then click [OK]

Localization – Steps (4/5)

- Localization



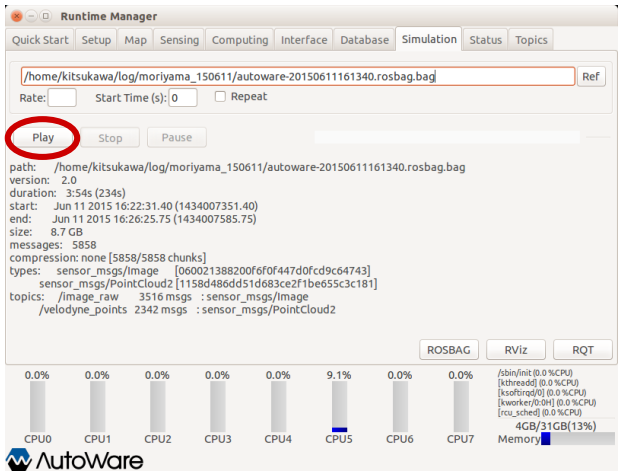
6. Localization

A) Launch [ndt_matching in [Computing] tab by checking ☑ the box

B) Click [app], and specify [Initial Pos] (e.g., all 0) in the displayed window

C) Click [OK]

- ➢ Initial localization position can be specified on [2D Pose Estimate] in [Rviz]

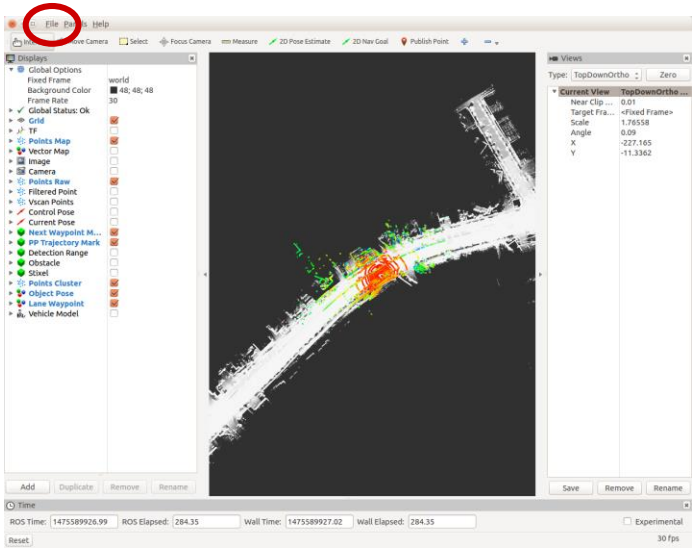


7. ROSBAG playback

A) Click [Play] button in [Simulation] tab

Localization – Steps (5/5)

- Localization



8. Displaying the result on RViz
   - A) Launch Rviz, and then specify "Autoware/ros/src/.config/rviz/default.rviz" in [File]->[Open Config]
   - B) Confirm that the map and the scan data are overlapped

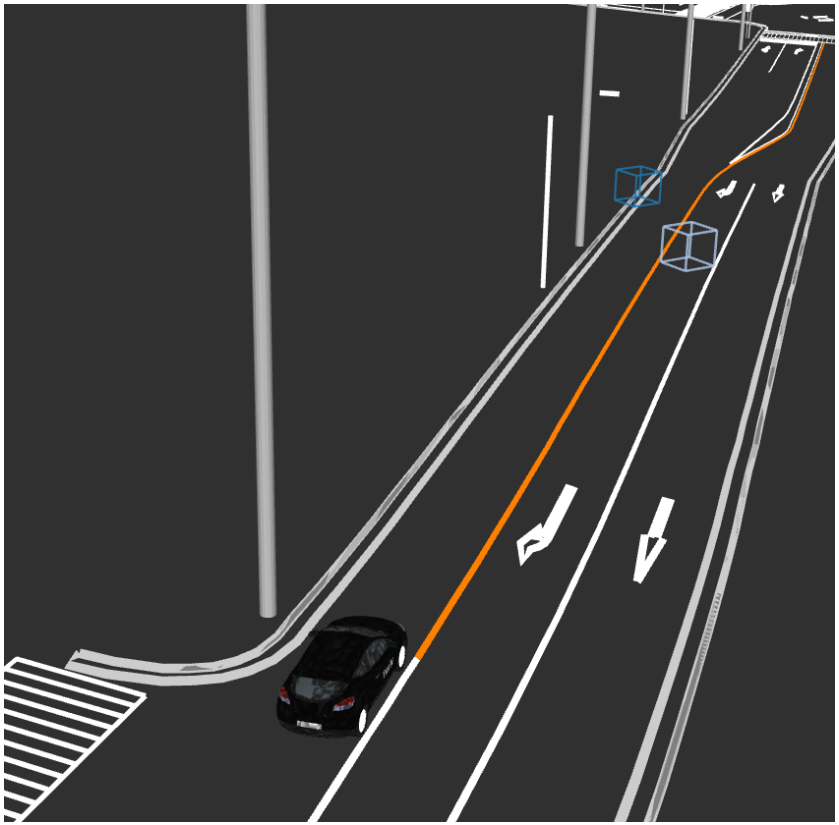
Autoware Hands-on Exercises

Chapter 3 : Object Detection and Traffic Light Detection

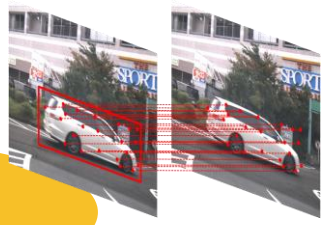
1. Object Detection

Object Detection – Overview

- Detect vehicles and pedestrians from camera images
- Calculate the distances to the detected objects using LIDAR point information
- Calculate the 3D coordinates of the detected objects by Reprojection



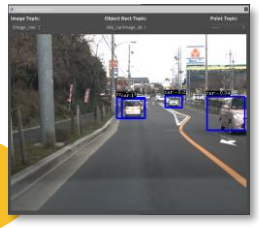
Object Detection – Structure



※ Some topics are omitted

/image_raw

Camera image



/dpm

Detect objects from the image

/image_obj

/points_image

/points2image

Project LIDAR points on the image

/tracking

Associate detected objects with similar objects, and track them

/image_obj_ranged

/image_obj_tracked

/projection_matrix

/tf

/reprojection

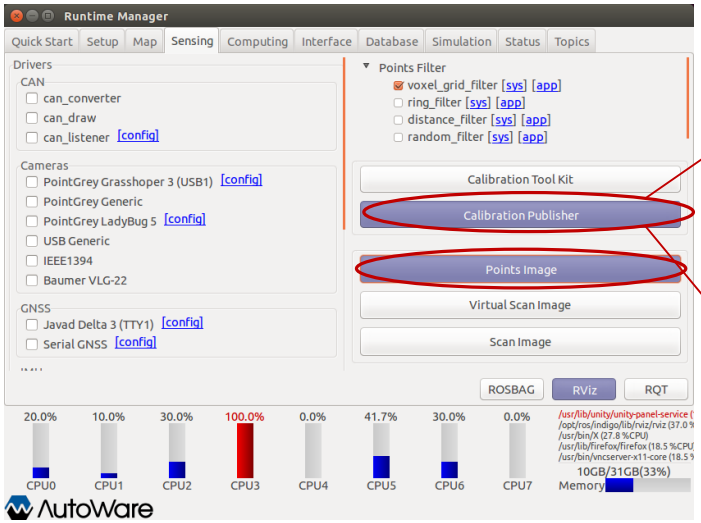
Calculate the 3D coordinates of the detected objects

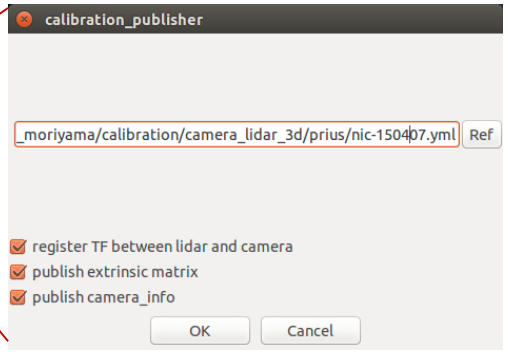
/obj_label_ bounding_box

/ranging

Calculate the distances to the detected objects

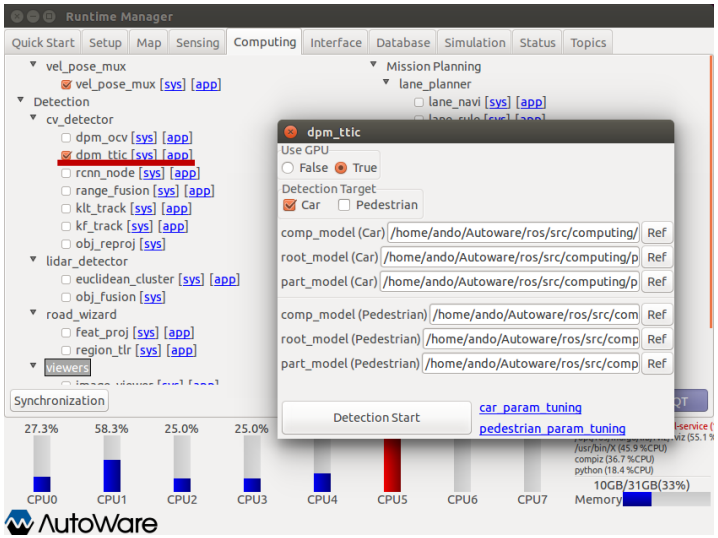
Object Detection – Step (1/4)





1. Calibration file loading
   - A) Click [Calibration Publisher] button in [Sensing] tab
   - B) Specify a calibration file on the displayed file selection window
2. Points Image launching
   - A) Click [Points Image] button

Object Detection – Step (2/4)



3. dpm_ttic launching

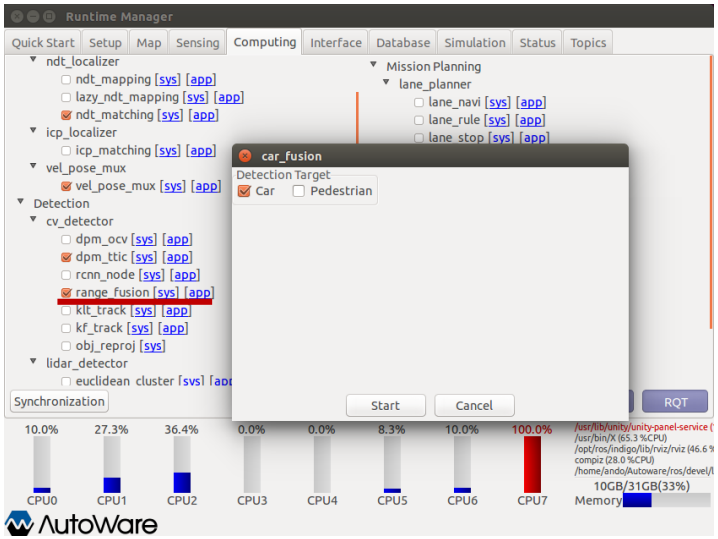
3. "dpm_ttic" launching

A) Check the box of [dpm_ttic], specify parameters as the left-top figure on the displayed window, and then press [Detection Start]

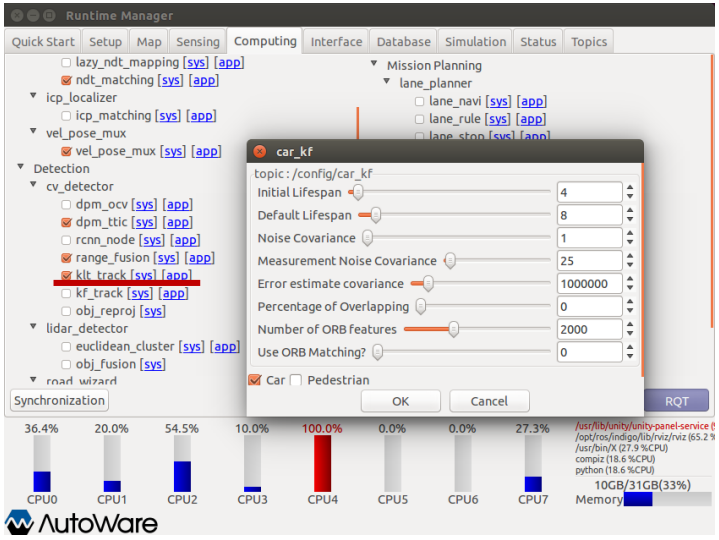
4. "range_fusion" launching

A) Check [range_fusion], press [Start] button on the displayed window

5. "klt_track" launching

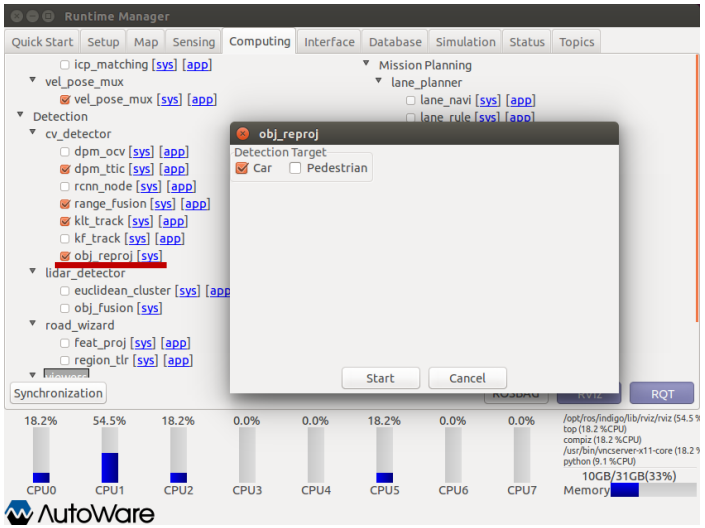


4. range_fusion launching



5. klt_track launching

Object Detection – Step (3/4)



6. Launching opj_reproj

6. "opj_reproj" launching

A) Check [opj_reproj], and then press [Start] button

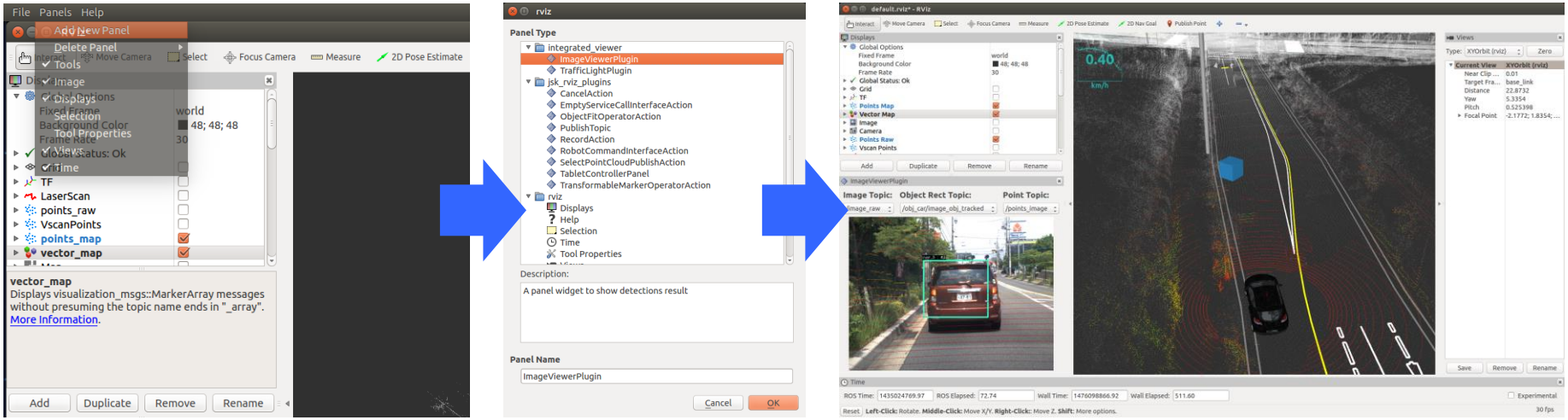
7. A window panel addition for displaying detection result

A) Select [Panels]→[Add new panel] in RViz

B) Select [Image Viewer Plugin]

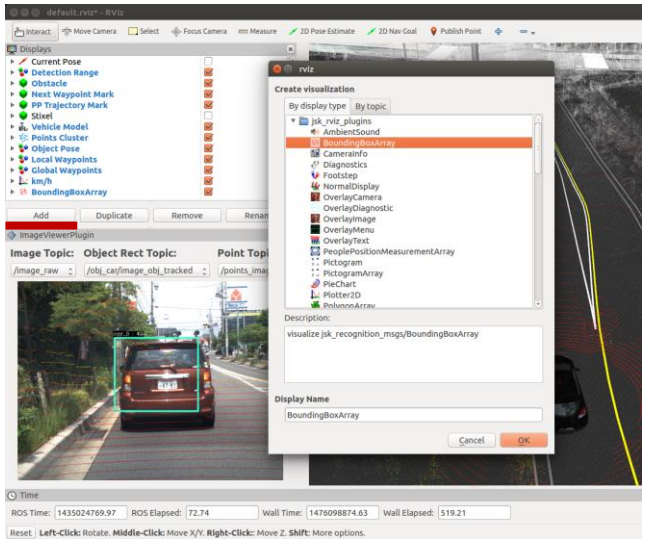
C) Adjust the panel to prefer position and size

D) Specify [Image Topic:] on [ImageViewerPlugin] as shown below:



7. Adding detection results window panel

Object Detection – Step (4/4)



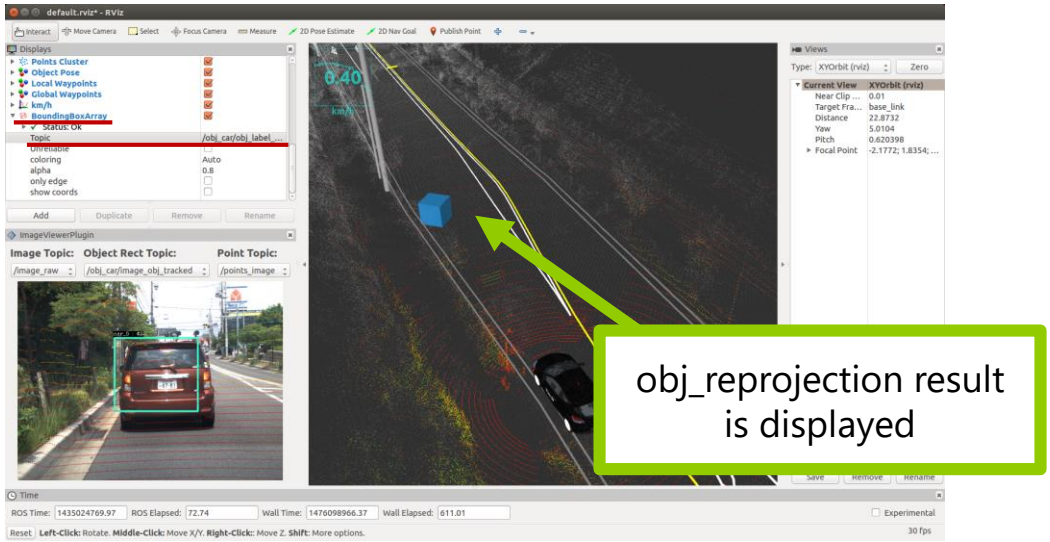
8. (A)(B) obj_reprojection result displaying

8. "obj_reprojection" result displaying

A) Press [Add]

B) Select [BoundingBoxArray]

C) After [BoundingBoxArray] is added to the left top list in Rviz, select "/obj_car/obj_label_bounding_box" in [Topic]



8. (C) obj_reprojection result displaying

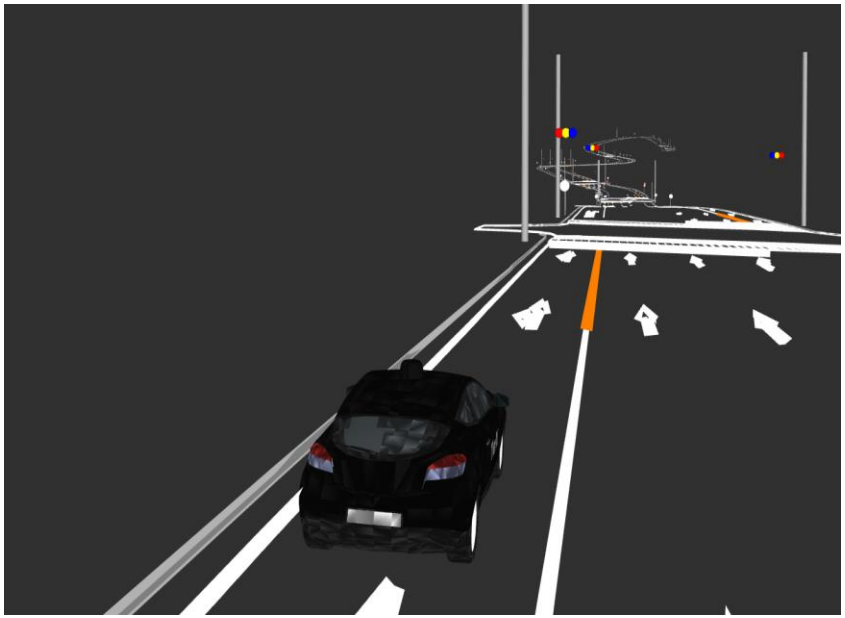
Autoware Hands-on Exercises

Chapter 3 : Object Detection and Traffic Light Detection

2. Traffic Light Detection

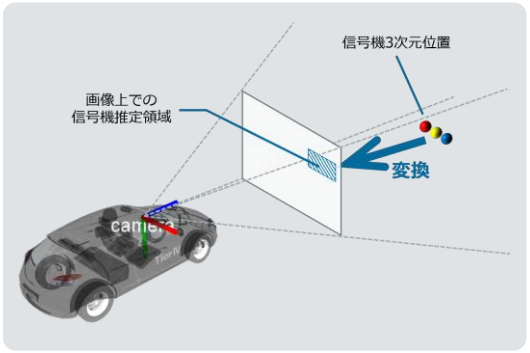
Traffic Light Detection – Overview

- Detects traffic light color from camera images
- Calculates the coordinates of traffic lights from the current position and VectorMap information obtained by localization
- Starting/Stopping at traffic light can be enabled by linking the detection result with a path planning node



Traffic Light Detection – Structure

※ Some topics are omitted



/vector_map_info

/tf

/feature_projection

Extracts traffic light positions
from TF between the Vector Map
the camera-LIDAR

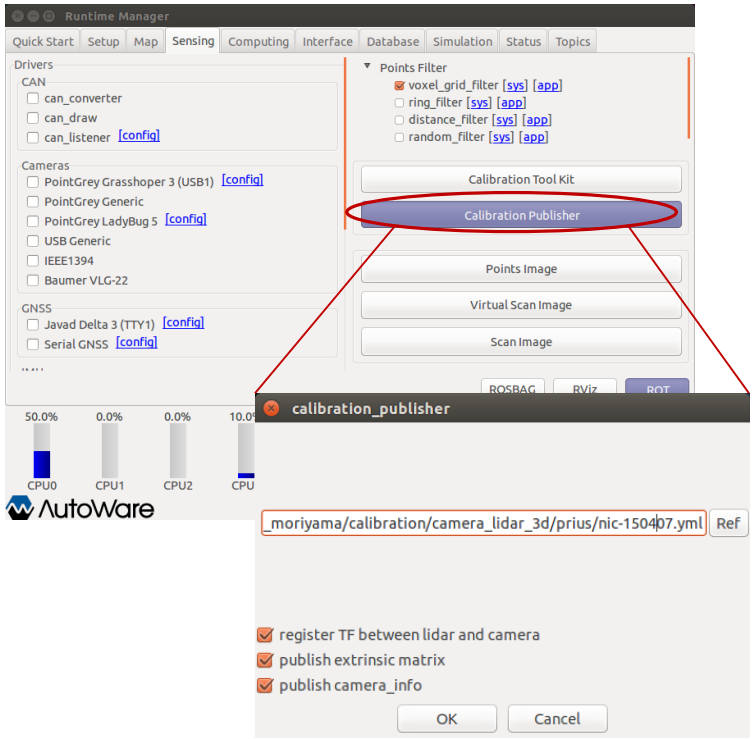
/roi_signal

/traffic_light_
recognition

Recognizes traffic the light colors
from the ROI values

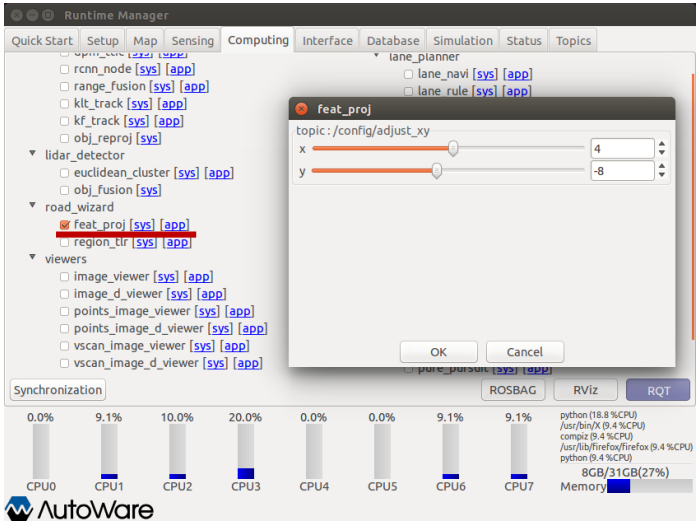
/light_color

Traffic Light Detection – Step (1/3)

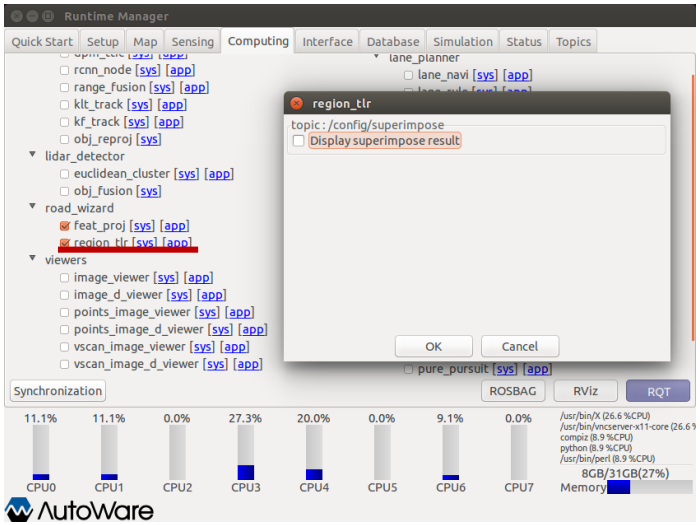


1. A calibration file loading
   A) Click [Calibration Publisher] button in [Sensing] tab
   B) Select a calibration file in the displayed file-selection-window

Traffic Light Detection – Step (2/3)



2. feat_proj launching



3. region_tlr launching

2. "feat_proj" launching

A) launch [feat_proj] in [Computing] tab by checking ☑ the box

➢ If Calibration is not precise enough, the detection positions can manually be corrected by [app]

3. "region_tlr" launching

A) launch [region_trl] by checking ☑ the box

Traffic Light Detection – Step (3/3)

4. Window panel addition for displaying detection results

A) Select [Panels] – [Add new panel] on RViz

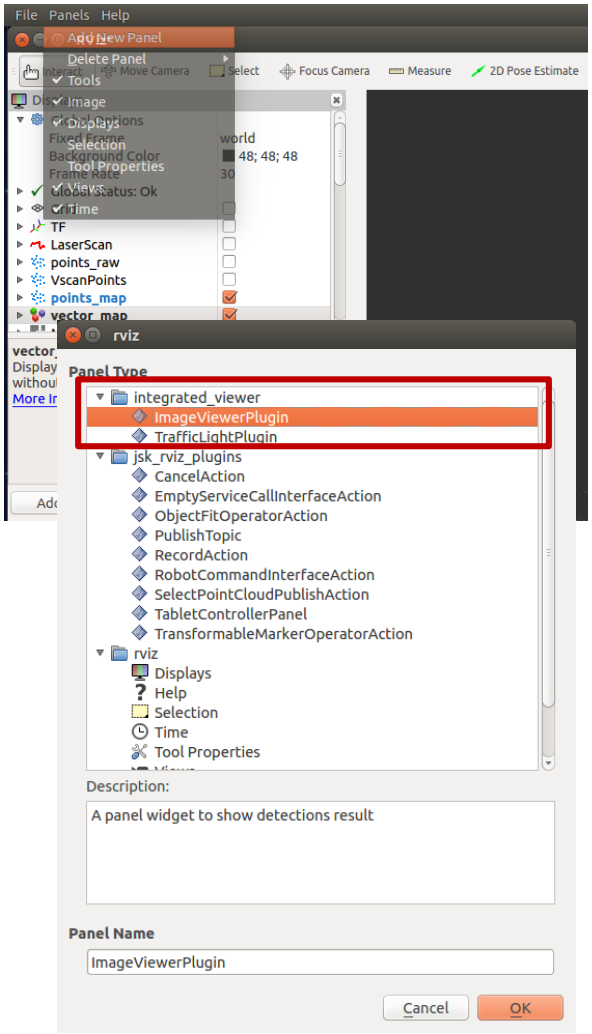
B) Panel list will be shown, and select [Image Viewer Plugin] and [Traffic Light Plugin]

C) Adjust panel to prefer position and size

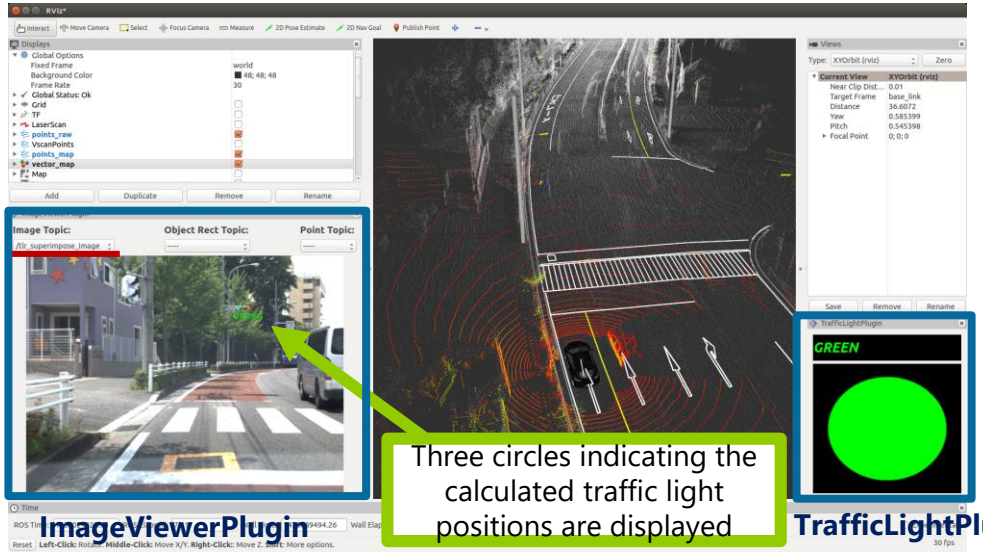
D) Change [Image Topic:] on [ImageViewerPlugin] to [/tlr_superimpose_image]

5. Traffic light detection position adjusting

A) If there is displacement between the three displayed circles and the traffic light positions, adjust [feat_proj] with [app]



4. Window panel addition for displaying detection results



Autoware Hands-on Exercises

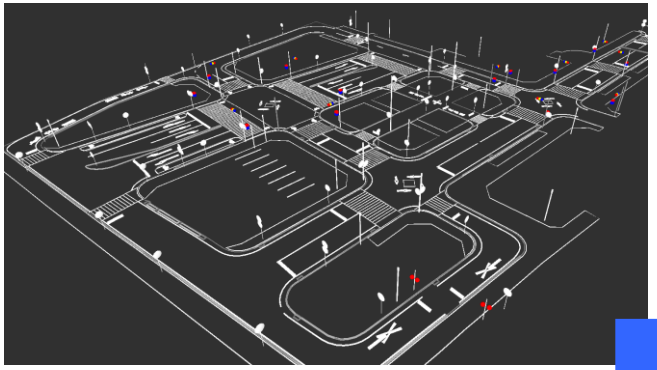
Chapter 4：Path Generation and Path Planning

1. Path Generation

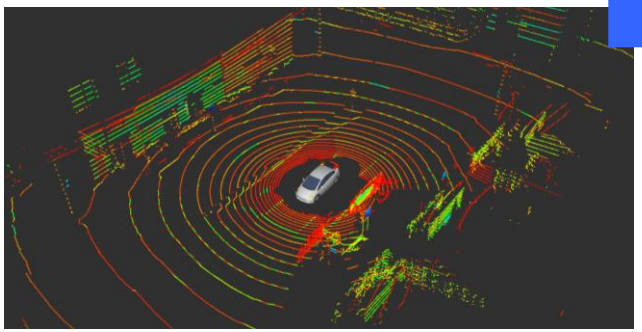
Path Generation – Overview

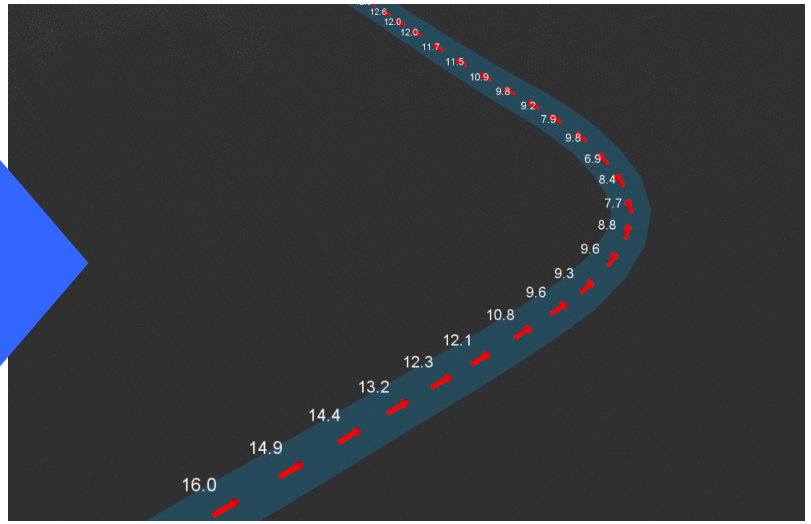
- From Vector Map and localization information, a path is generated
- If Vector Map is used, a path is generated from the centerline information the road
- If no Vector Map is available, a path is generated from log data
- The path is output as data string with coordinate, direction and velocity information

Vector Map



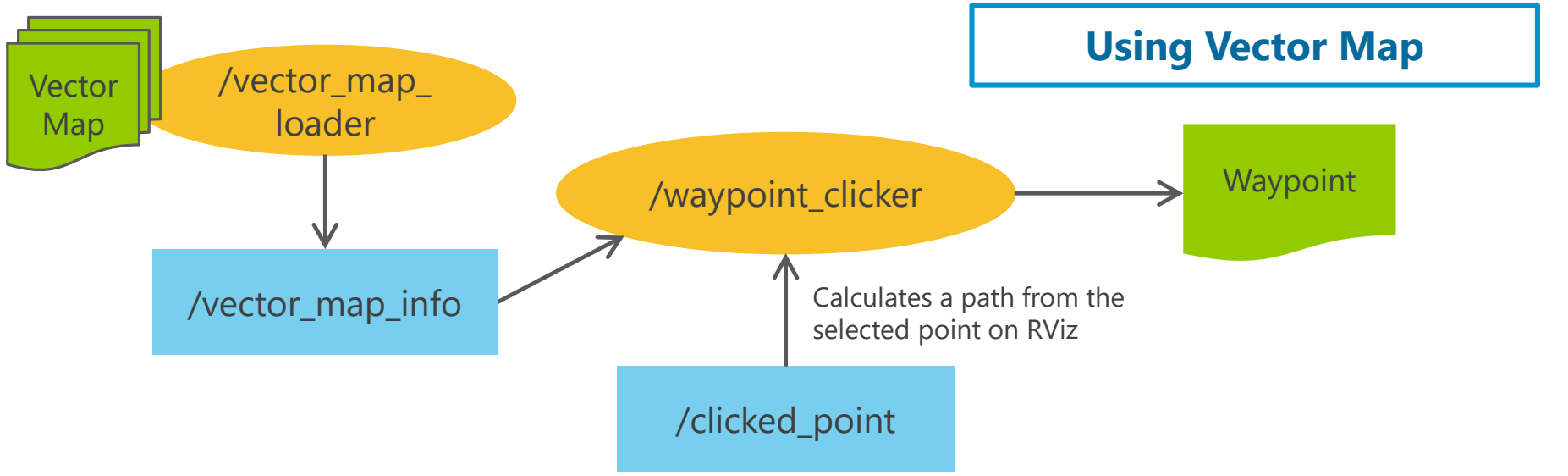
Position and velocity information from localization

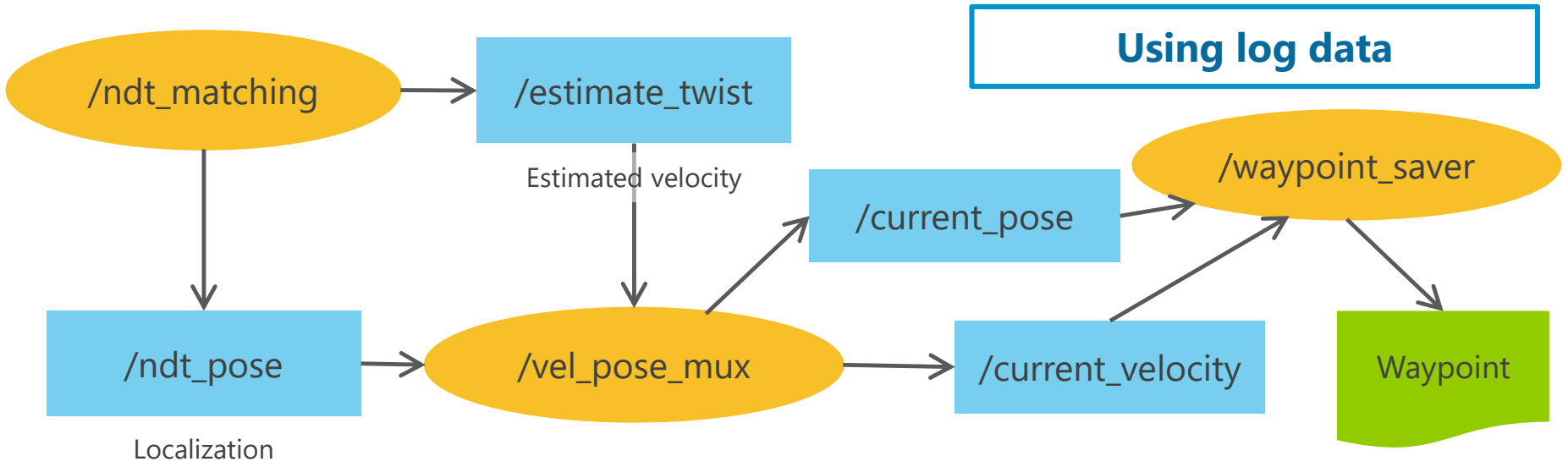




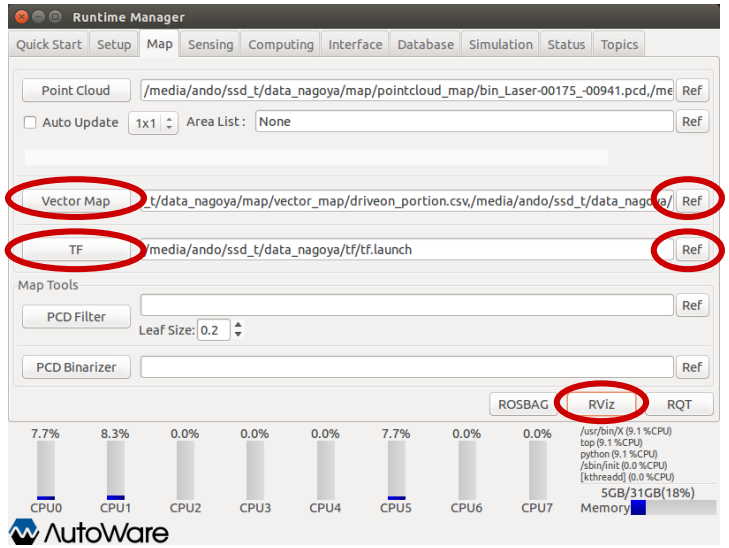
Path (Waypoint)

Path Generation – Structure





Path Generation Using Vector Map – Step (1/2)

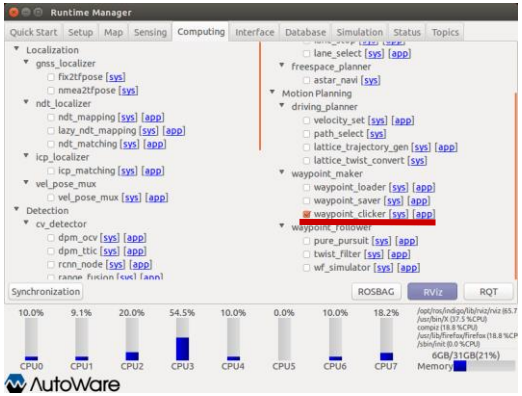


1. Vector Map loading
   A) Clicking [Ref] on [Vector Map] in [Map] tab, specify a Vector Map file
   B) Click [Vector Map] button

2. Specify TF to indicate Vector Map position
   A) Clicking [Ref] on [TF] in [Map] tab, select a TF file according to the Vector Map position.
   B) Click [TF] button
   (this publishes TF between world frames and map frames)

3. Launch RViz
   A) Click [Rviz]
   B) Pressing either [File]->[Open Config], or [Ctrl + O], specify [Autoware/ros/src/.config/rviz/default.rviz]
   C) Confirm that Vector Map is displayed

Path Generation Using Vector Map – Step (2/2)



4. waypoint_clicker launching

5. Path generation by [Publish Point] in RViz

4. Launch “waypoint_clicker”

A) Launch [waypoint_clicker] checking (☑) the box in [Computing] tab

➢ Path, velocity and path to save the file can be specified

5. Path generation by [Publish Point] on RViz

A) Confirm that multiple yellow points are being displayed on RViz

B) Click [Publish Point] in the top of RViz, click the yellow points displayed on RViz

➢ Zooming and moving the mouse over the yellow points in RViz, these points can be clicked.

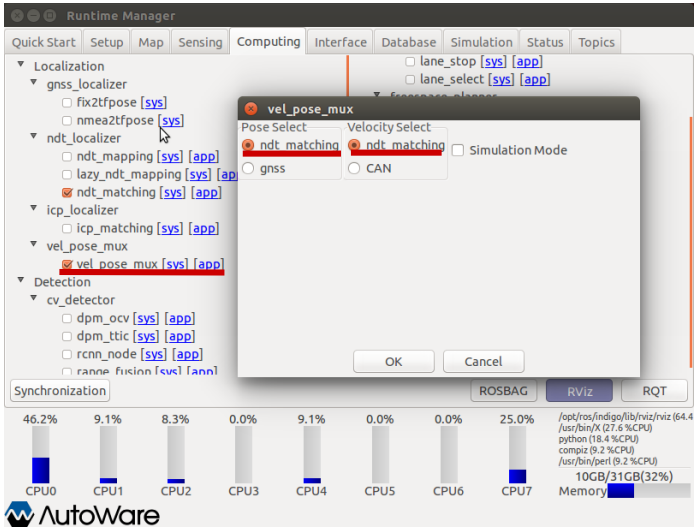
C) Clicking the yellow points on the same lane, yellow line is generated

➢ If the selected yellow points are linkable, even long paths can be generated

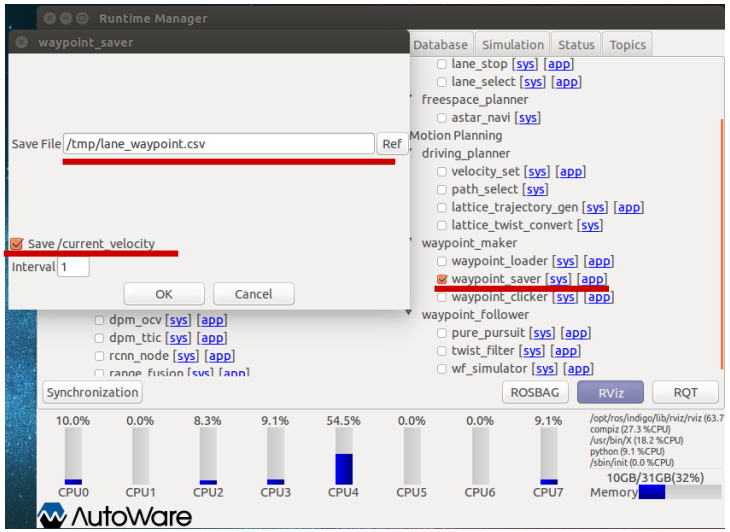
➢ Note that, if green or red points are located on intersections, path generation may not success without clicking them

D) Checking off the box of [waypoint_clicker] in [Computing] tab, a path file is generated in the directory specified in [app]

Path Generation by ROSBAG – Step (1/2)



2. vel_pose_mux launching



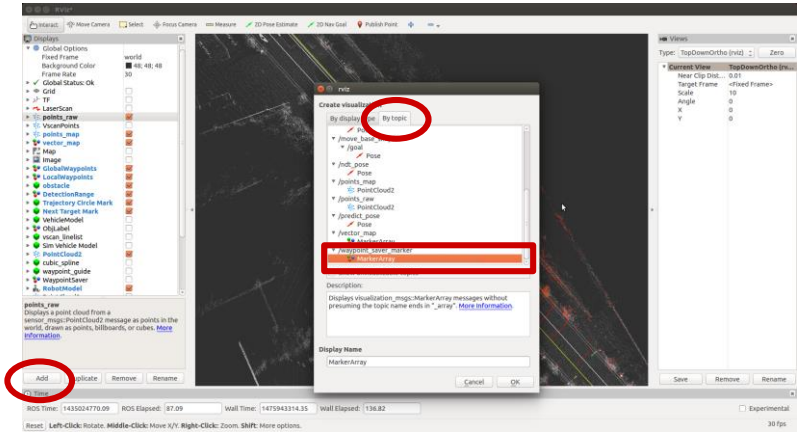
3. waypoint_saver launching

1. ROSBAG playback and localization
   - ➢ Refer to localization

2. “vel_pose_mux” launching
   - A) Launch [vel_pose_mux] by checking (☑) the box in [Computing] tab
   - ➢ Confirm that the box of [Simulation Mode] in [app] is not checked, and [ndt_matching] of both [Pose Select] and [Velocity Select] are checked on.

3. “waypoint_saver” launching
   - A) Launch [waypoint_saver] by checking (☑) the box in [Computing] tab

Path Generation by ROSBAG – Step (2/2)

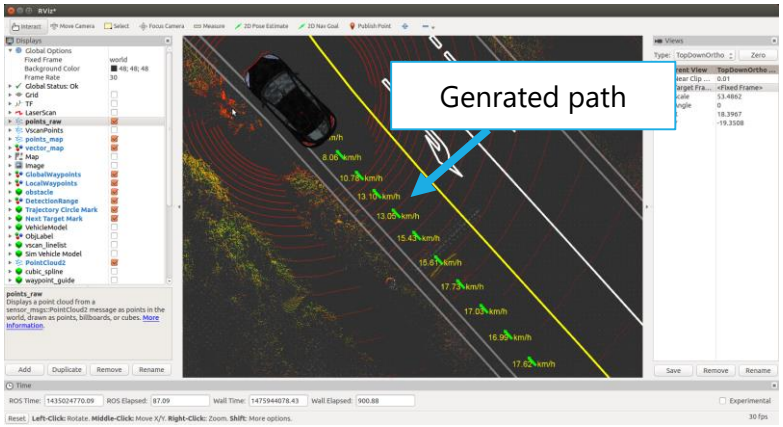


4. Displaying “/waypoint_saver_marker” on RViz

A) Click [add] button in the bottom of the left topic list in RViz

B) Click [/waypoint_saver_marker] -> [MarkerArray] from [By Topic] tab of the displayed window.

C) Confirm the marker logging waypoint in RViz is being displayed



5. Ending “waypoint_saver”

A) End path generation by unchecking off [waypoint_saver] on [Computing] tab in [Runtime Manager]

➢ The path file is saved into the directory specified in [app]

4. /waypoint_saver_marker on RViz displaying

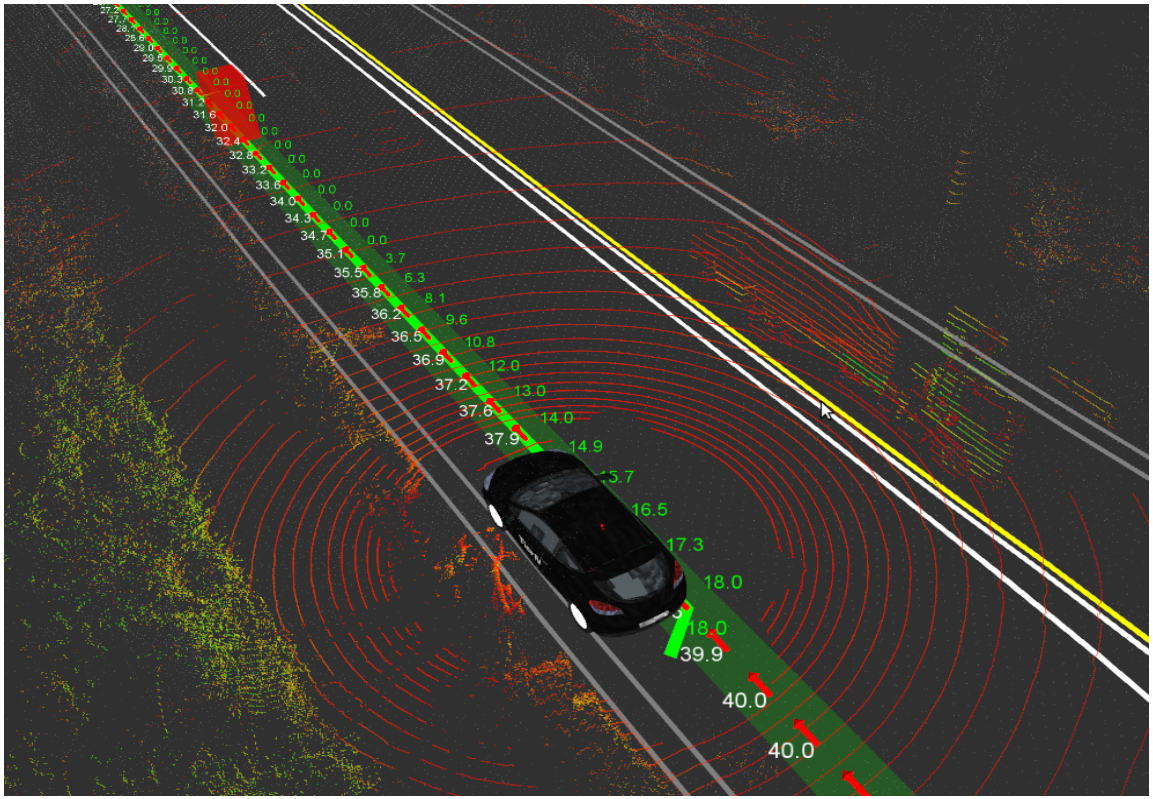
Autoware Hands-on Exercises

Chapter 4：Path Generation and Path Planning

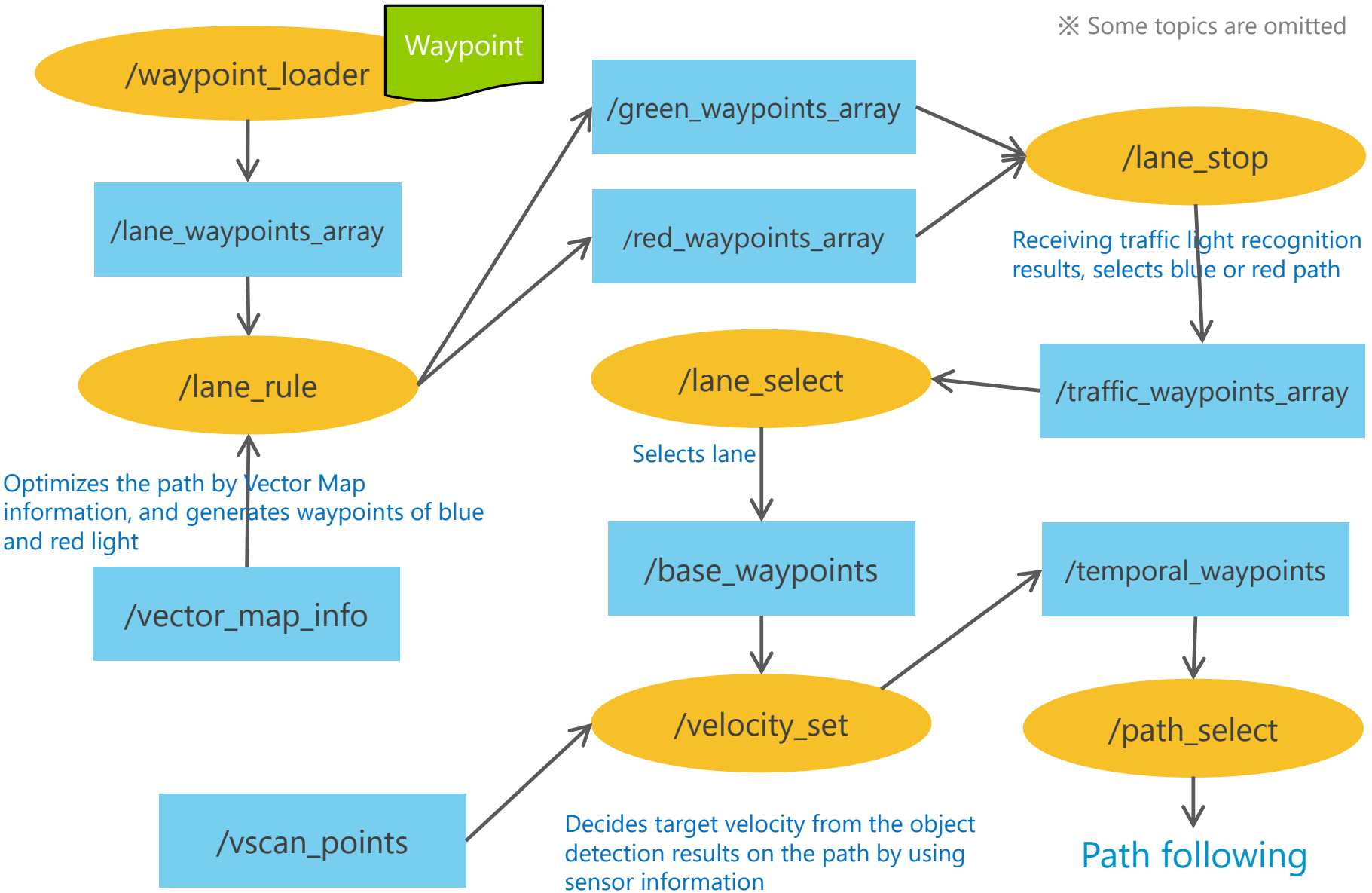
2. Path Planning

Path Planning – Overview

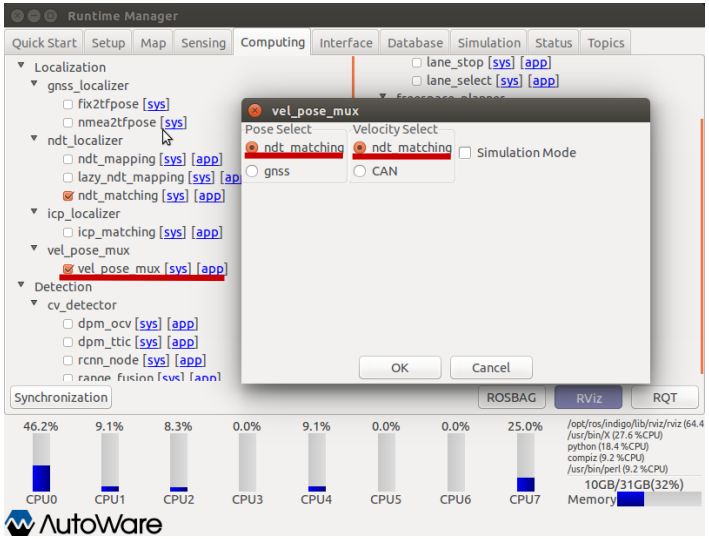
- Loads a previously saved path file
- Optimizes velocities on the path with high definition map information, etc.
- Selects a lane, if there are multiple lanes



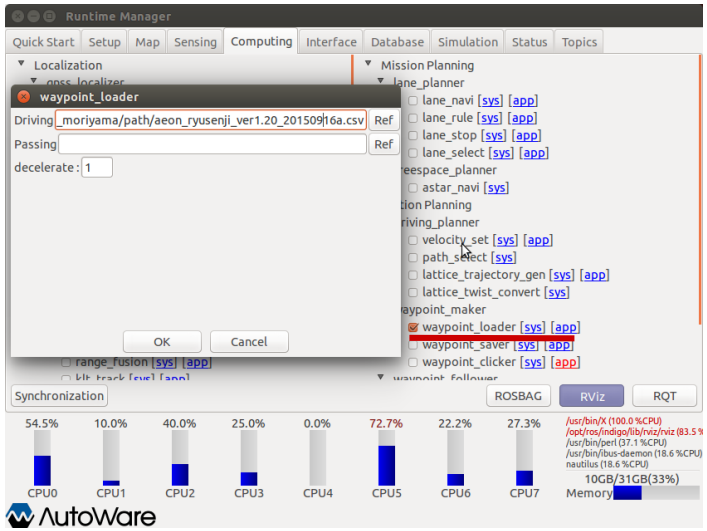
Path Planning – Structure



Path Planning – Step (1/4)



2. vel_pose_mux launching



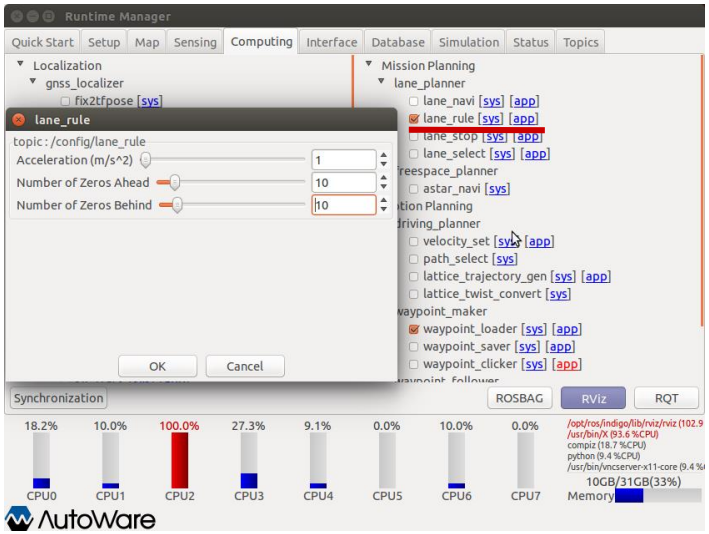
3. waypoint_saver launching

1. ROSBAG playback and localization
   - ➢ Refer to localization chapter

2. "vel_pose_mux" launching
   - A) Launch [vel_pose_mux] in [Computing] by checking (☑) the box
     - ➢ Confirm that [Simulation Mode] in [app] is unchecked and [ndt_matching] of both [Pose Select] and [Velocity Select] are checked

3. Path loading using waypoint_loader
   - A) Specify the path file in [app]
     - ➢ Two paths can be specified. This loader can launch only one path

Path Planning – Step (2/4)

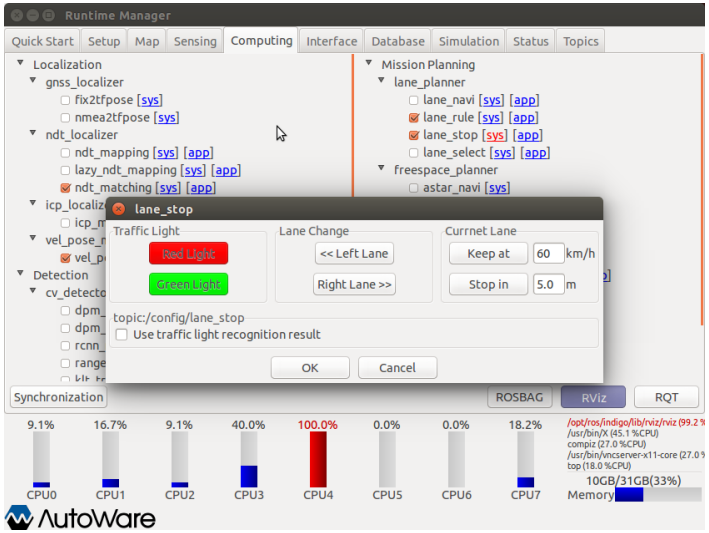


4. lane_rule launching

4. Launching “lane_rule”

A) Launch [lane_rule] by checking (☑) the box

- [Number of Zeros Ahead] and [Number of Zeros Behind] define the number of waypoint (velocity 0) at stop line when traffic light is red.



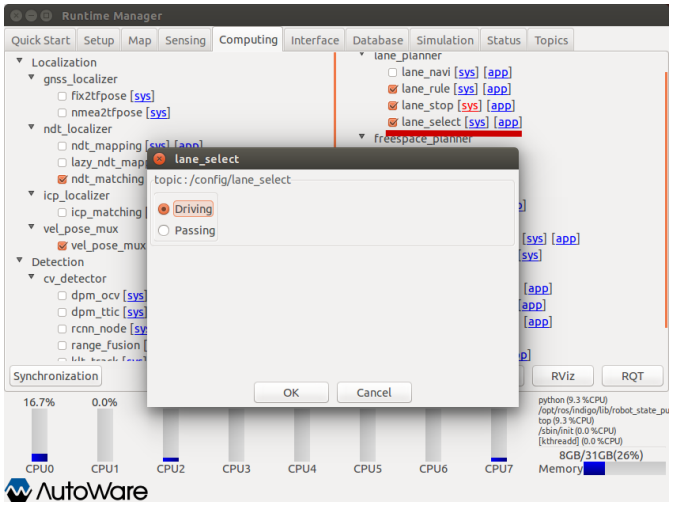
5. lane_stop launching

5. Launching “lane_stop”

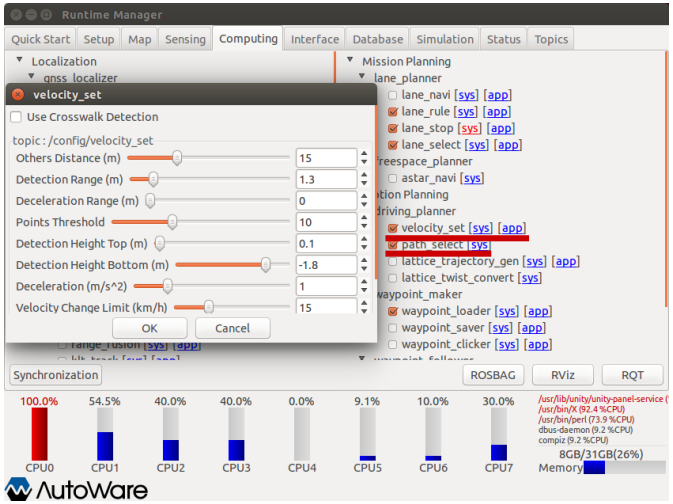
A) Launch [lane_stop] by checking (☑) the box

- Check [Use traffic light recognition], this node automatically switches path between blue and red path by receiving traffic light recognition node results.

Path Planning – Step (3/4)



6. lane_select launching



7. velocity_set launching
8. path_select launching

6. Launching "lane select"

A) Launch [lane_select] by checking (☑) the box

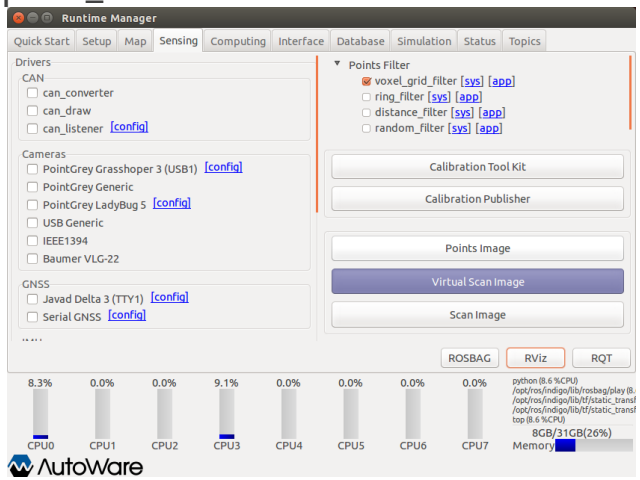
- Following paths can be selected in [app]
- If load only one path by [waypoint_loader], select the corresponding path.

7. Launching "velocity set"

A) Launch [velocity_set] by checking (☑) the box

- Refer left bottom figure for parameters
- To detect objects on the path, click [Virtual Scan Image] button in [Sensing] tab

8. Launching "path_select"

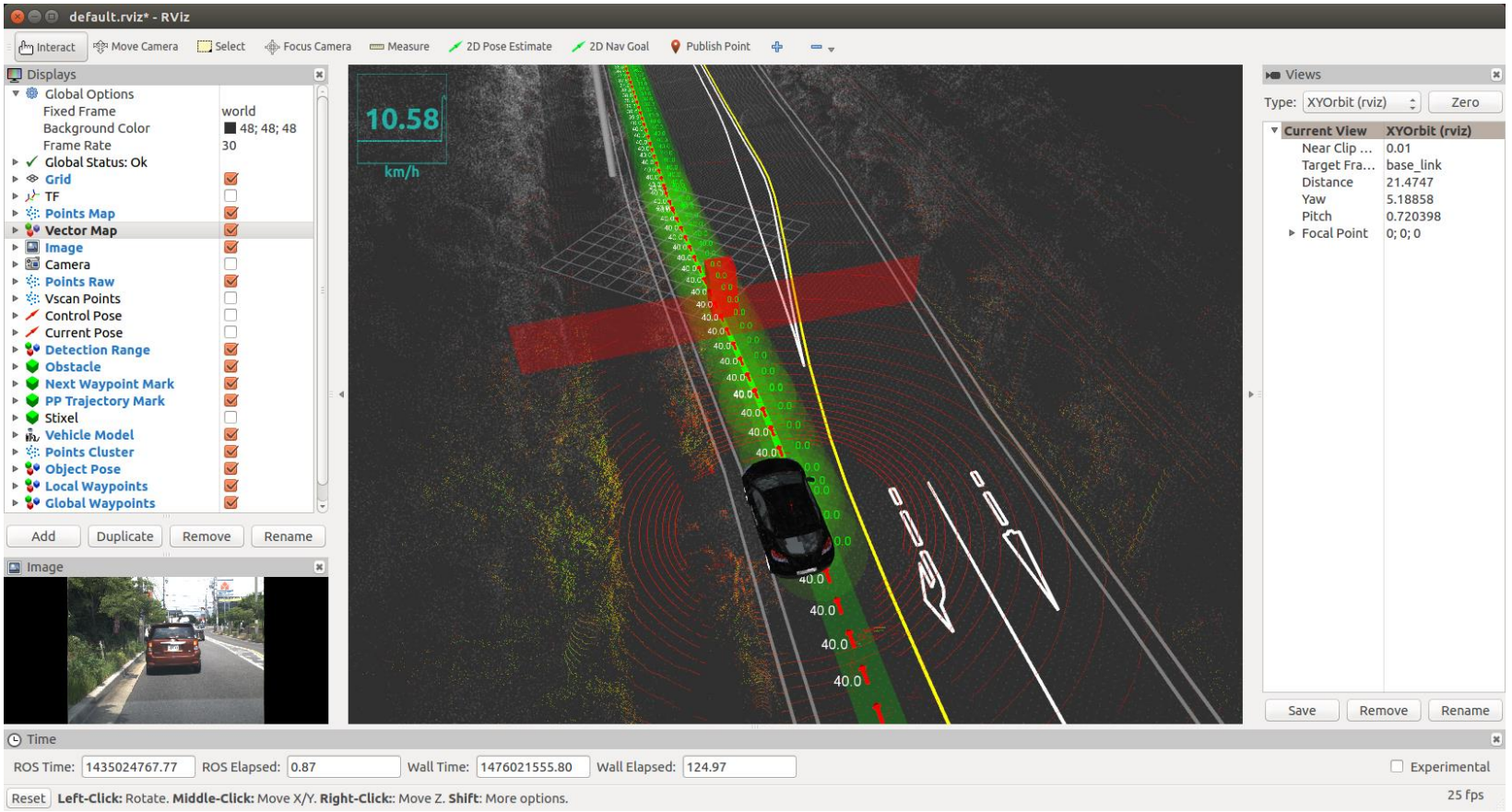


Virtual Scan Image launching

Path Planning – Step (4/4)

9. Confirm the path in RViz

- If light green path（Global Waypoints）and dark green path（Local Waypoints）are displayed, the nodes are working correctly.
- If some objects exist in front of the vehicle and object detection success, area boxes will be displayed as shown in the figure below.



Autoware Hands-on Exercises

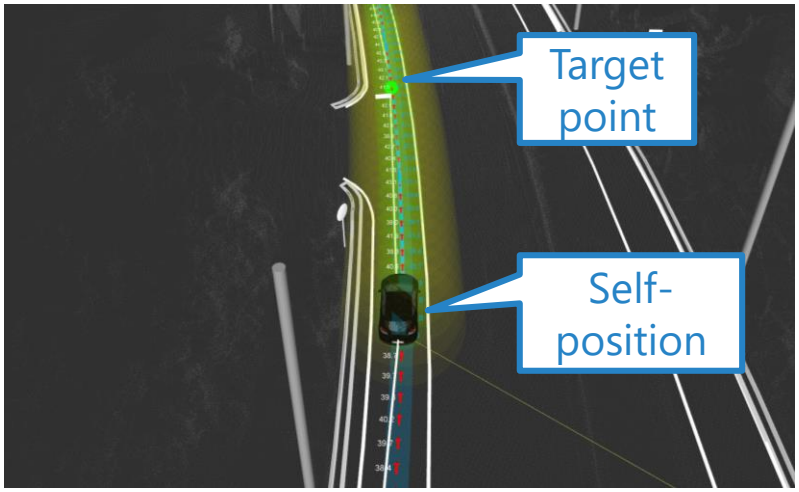
Chapter 5：Path Following and Vehicle Control

Path Following – Overview

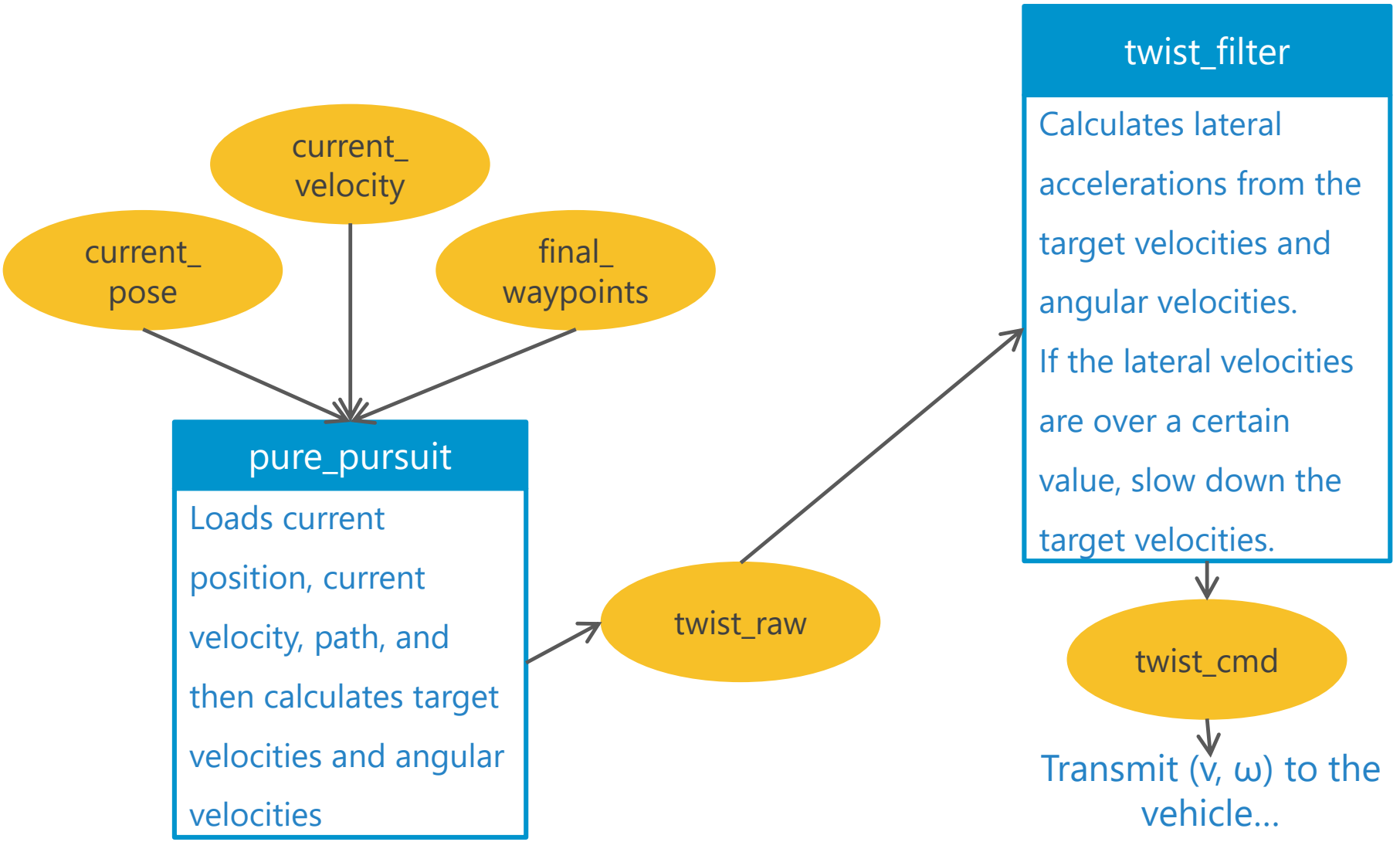
- Calculate vehicle control signals that the vehicle can follow
  - Vehicle control signals (v, ω): v…velocity, ω…angular velocity
  - Nodes: “pure_pursuit”, “twist_filter”
  - Subscribe topic: “/current_pose”, “/current_velocity”, “/final_waypoint”
  - Publish topic: “/twist_raw”, “/twist_cmd”

- Flow of path following and vehicle control

1. “pure_pursuit” node
   - Calculate the curvature of the circle passing through self-position and target points on the path
   - Calculate target angular velocities from the calculated curvature and the current velocity, and then publish them into a topic
2. “twist_filter” node
   - Calculate lateral accelerations from target velocities and angular velocities
   - If the lateral accelerations is over a certain value, slow down the target velocities and publish them into a topic
3. Transmit these nodes to the vehicle for control



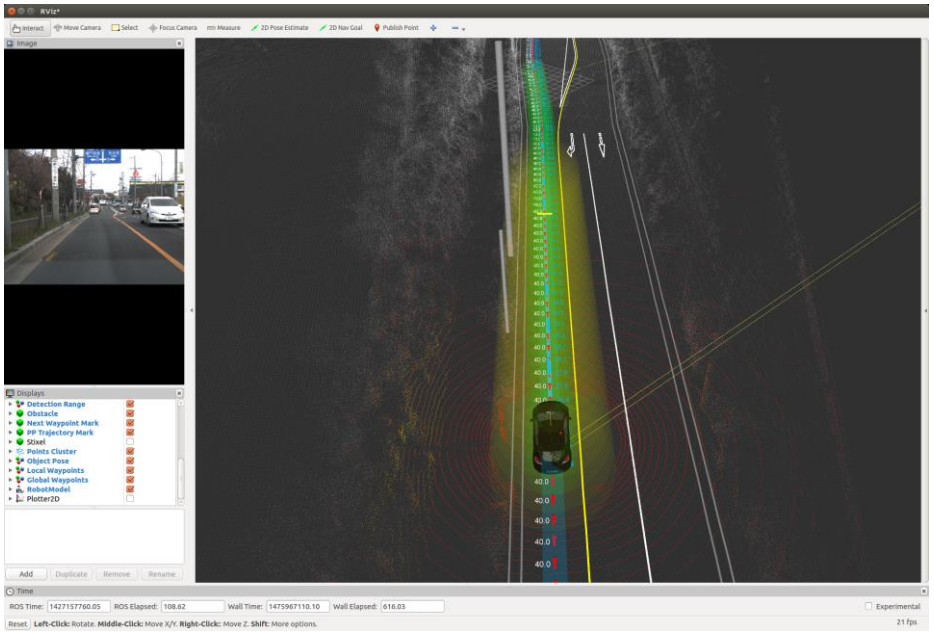
Path Following – Structure



Path Following – Steps in case of using ROSBAG (1/4)

- Assumptions
  - Localization has been completed, and "/current_pose", "/current_velocity" have been published
  - "/final_waypoints" has been published (path_select is launched)
  - ROSBAG is in Pause

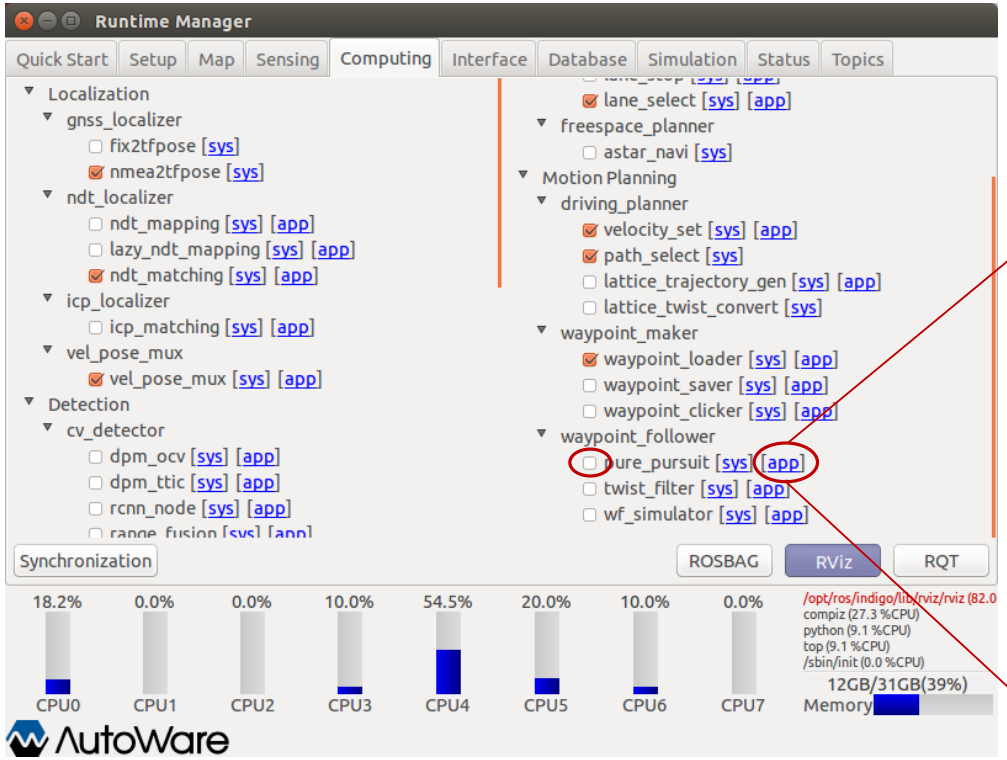
If the above assumptions are completed,
RViz displays somehing as shown below:

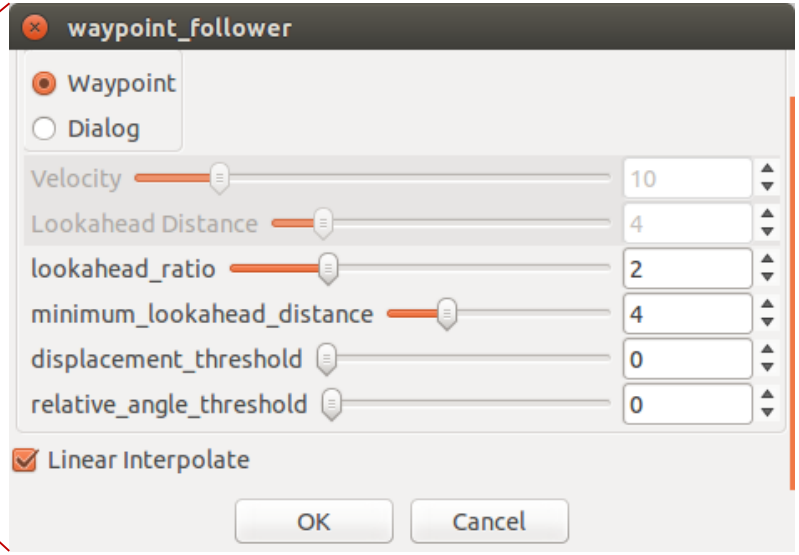


Path Following – Steps in case of using ROSBAG (2/4)

- Specify and launch “pure_pursuit”

1. Clicking [app] on “pure_pursuit”, a window will be displayed. Confirm that the values of the parameters are the same as the ones shown in the right-bottom figure
2. Launch pure_pursuit by checking (☑) the box

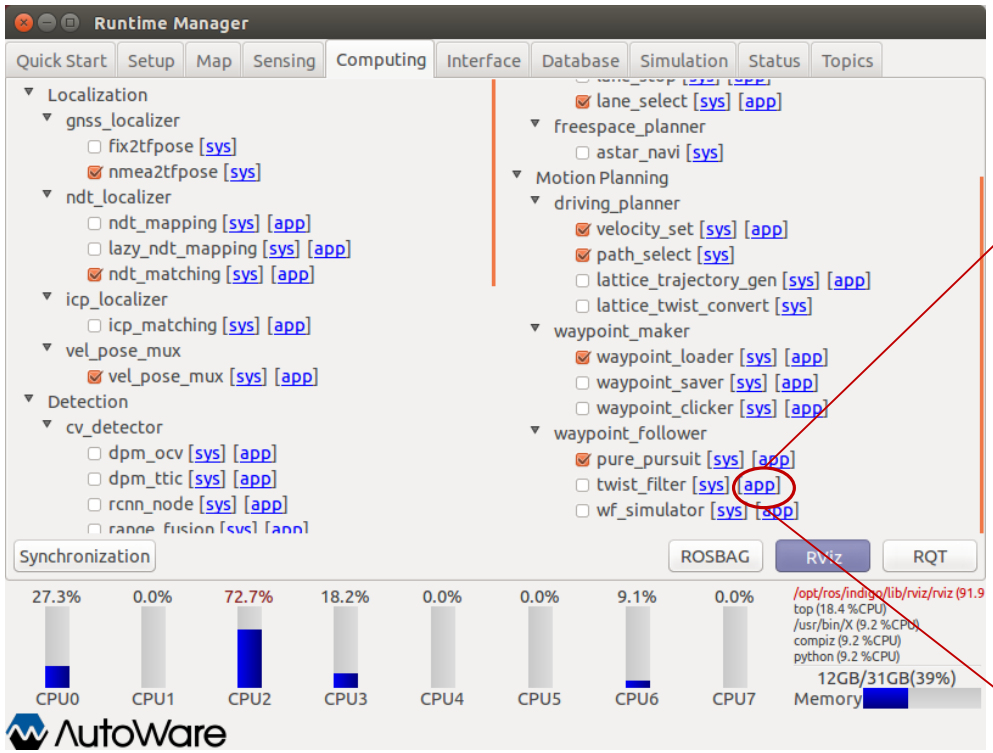


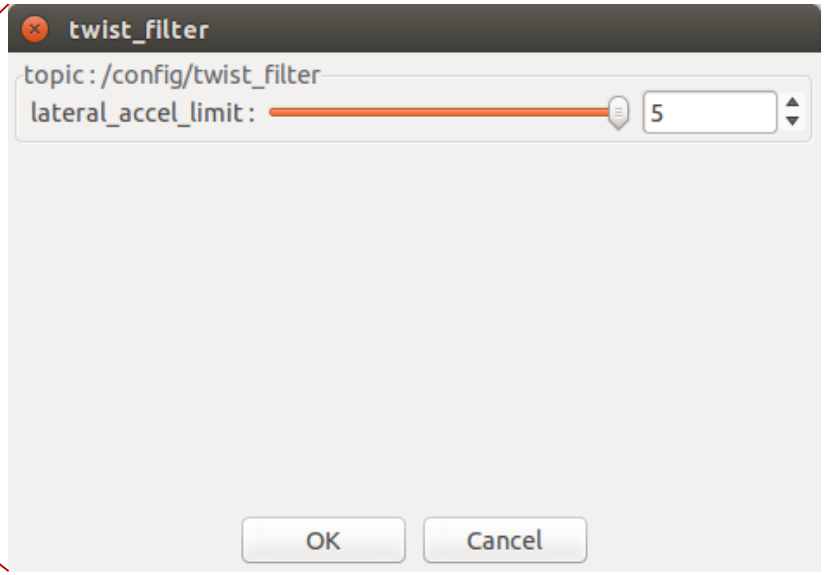


Path Following – Steps in case of using ROSBAG (3/4)

- Specify and launch “twist_filter”

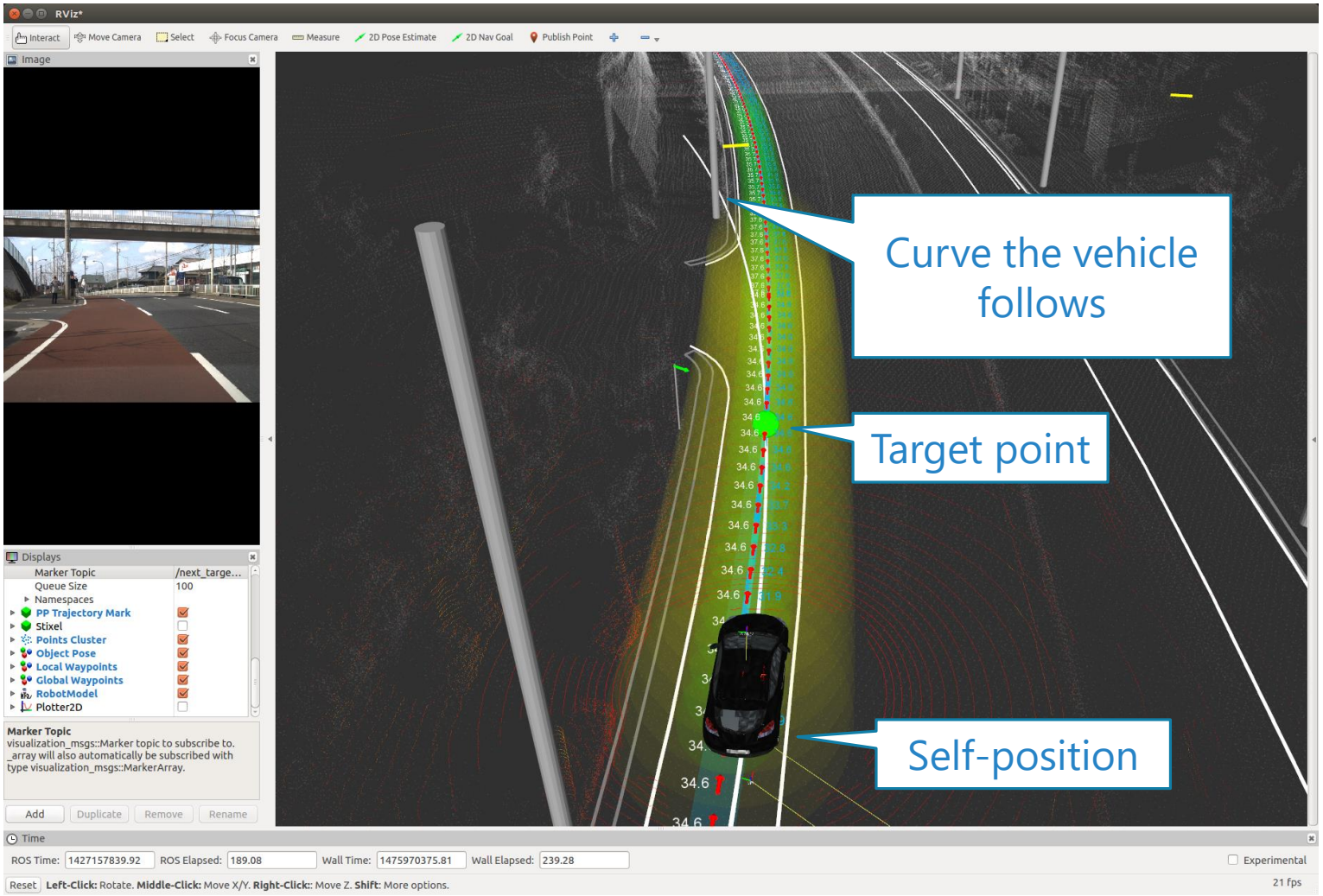
1. Clicking [app] on “twist_filter”, a window will be displayed. Confirm that the values of the parameters are the same as the ones shown in the right-bottom figure
2. Launch “twist_filter” by checking (☑) the box





Path Following – Steps in case of using ROSBAG (4/4)

- Resume playback using [Pause] of ROSBAG, path following results is displayed as shown below

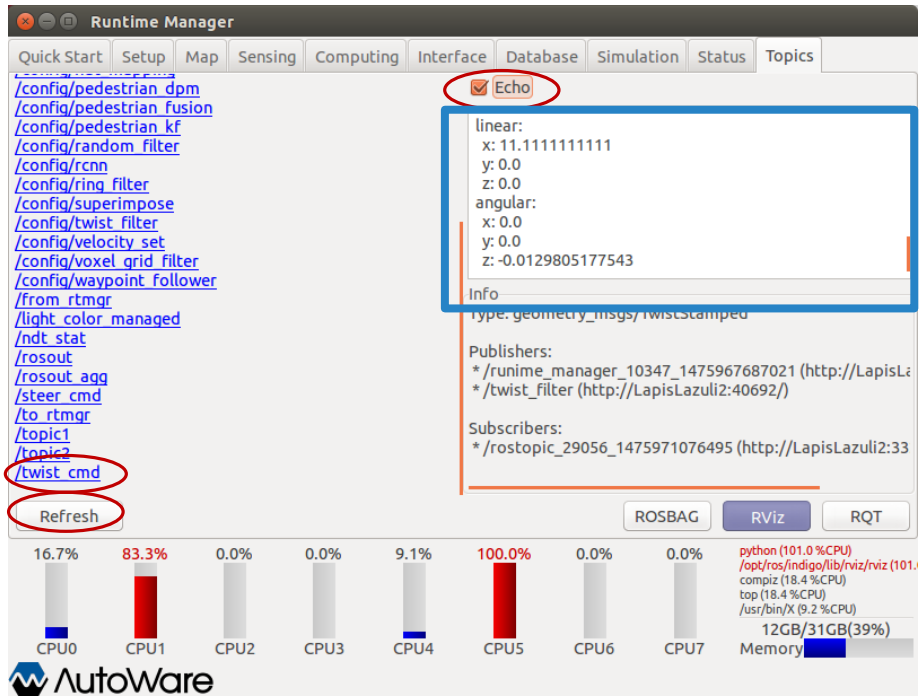


Path Following and Vehicle Control – Checking topics

- How to check published topics?

1. Check (☑) the box of [Echo] in [Topic] tab
2. Click "/[topic_name]" (here, "/twist_cmd" is clicked and displayed)
   - If the topic doesn't exist, press [Refresh]

After above steps,
the topic will be displayed inside the blue box



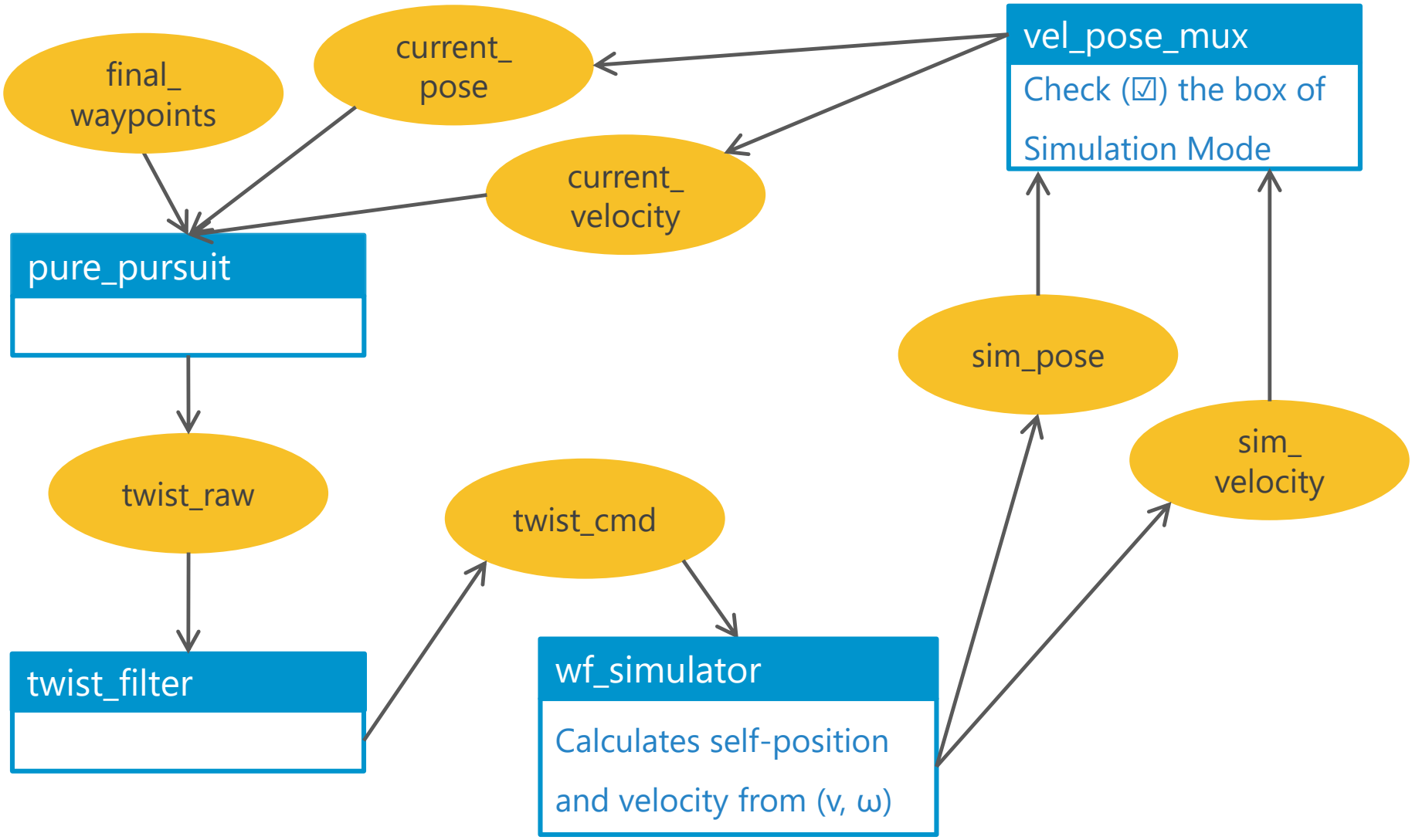
Path Following and Vehicle Control – in case of using wf_simulator

- What is “wf_simulator”?
  - Simulating an ideal self-position and velocity using the received vehicle control signals (v, ω)
  - The problem can be stated as:

$$x_{i+1} = x_i + v\cos\theta_i\,\Delta\mathrm{t}$$
$$y_{i+1} = y_i + v\sin\theta_i\,\Delta\mathrm{t}$$
$$\theta_{i+1} = \theta_i + \omega\Delta\mathrm{t}$$
$$i : 0, 1, \ldots, n$$

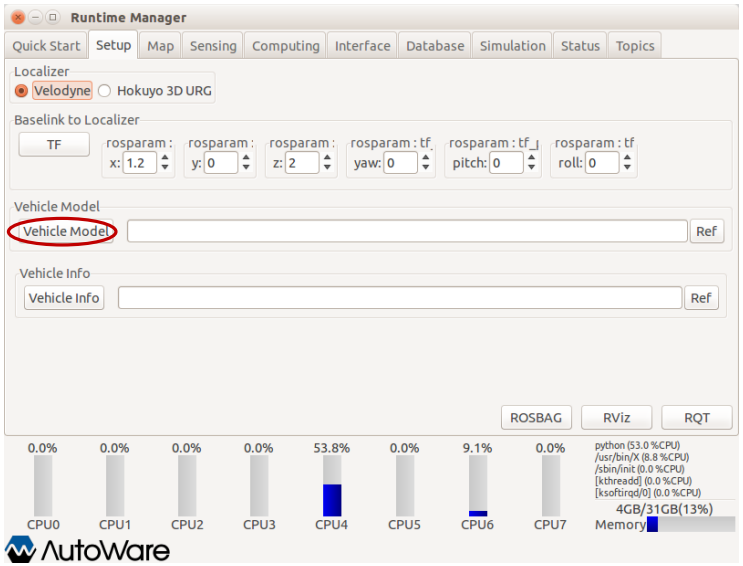
Path Following and Vehicle Control – in case of using wf_simulator

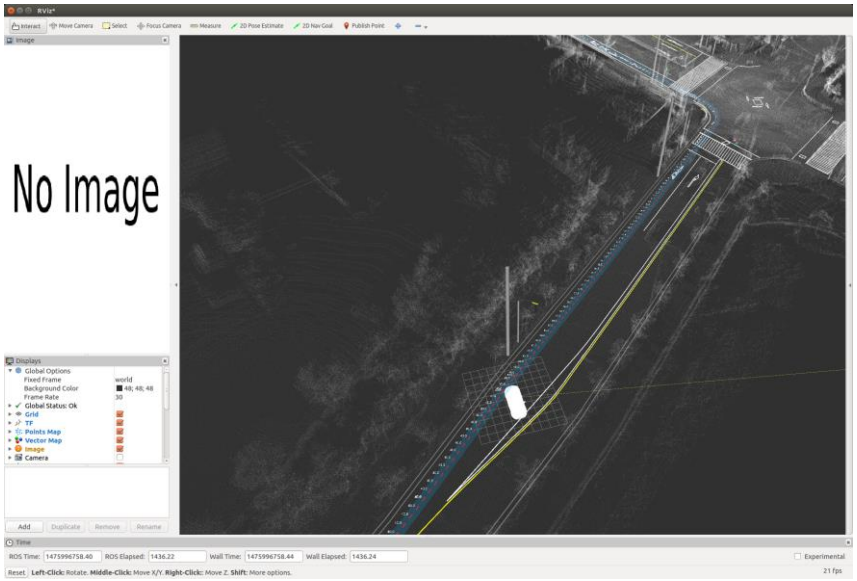
- Structure



Path Following and Vehicle Control – Steps of wf_simulator (1/6)

- Assumptions
  - Map and "/base_waypoints" have been published ("lane_select" has been launched)
  - If you have run some operations with ROSBAG previously, stop ROSBAG playing and restart TF and RViz
  - Load "/$HOME/Autoware/ros/.config/model/sim_default.urdf" on [Vehicle Model] in [Setup] tab

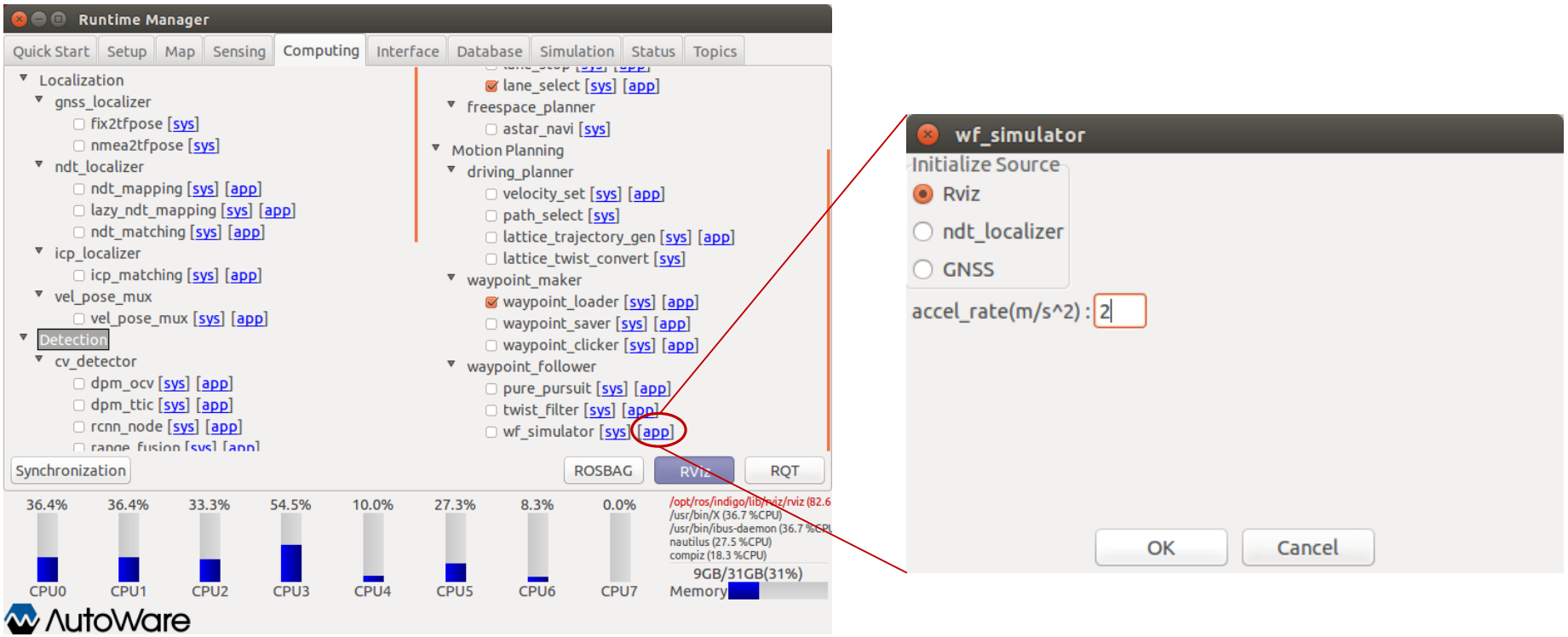




Path Following and Vehicle Control – Steps of wf_simulator (2/6)

- "wf_simulator" launching

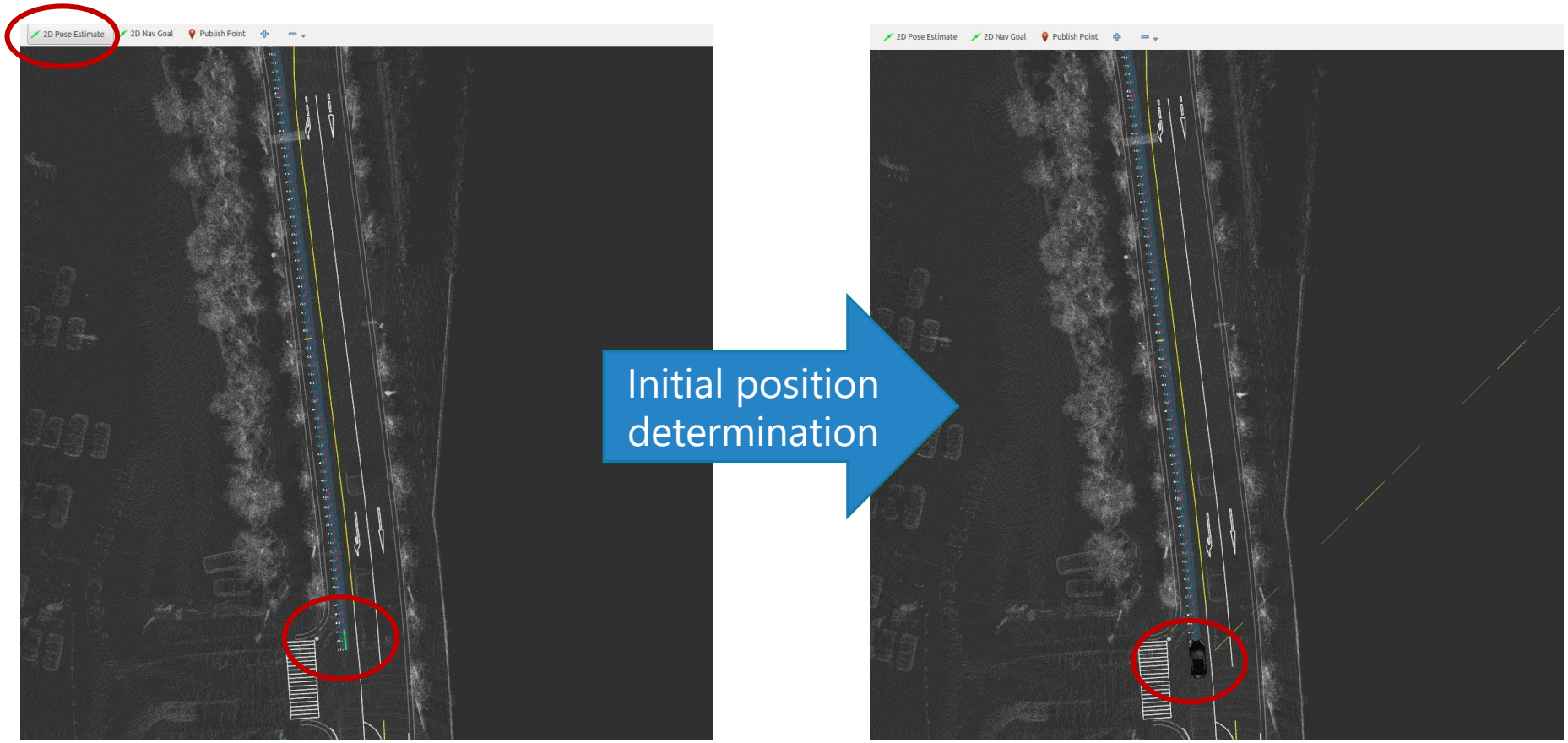
1. Clicking [app] on [wf_simulator], a figure like the one in the bottom-right will be displayed
   Confirm that the parameters are the same as the ones in the figure
2. Launch [wf_simulator] checking (☑) the box



Path Following and Vehicle Control – Steps of wf_simulator (3/6)

- Launching “wf_simulator”

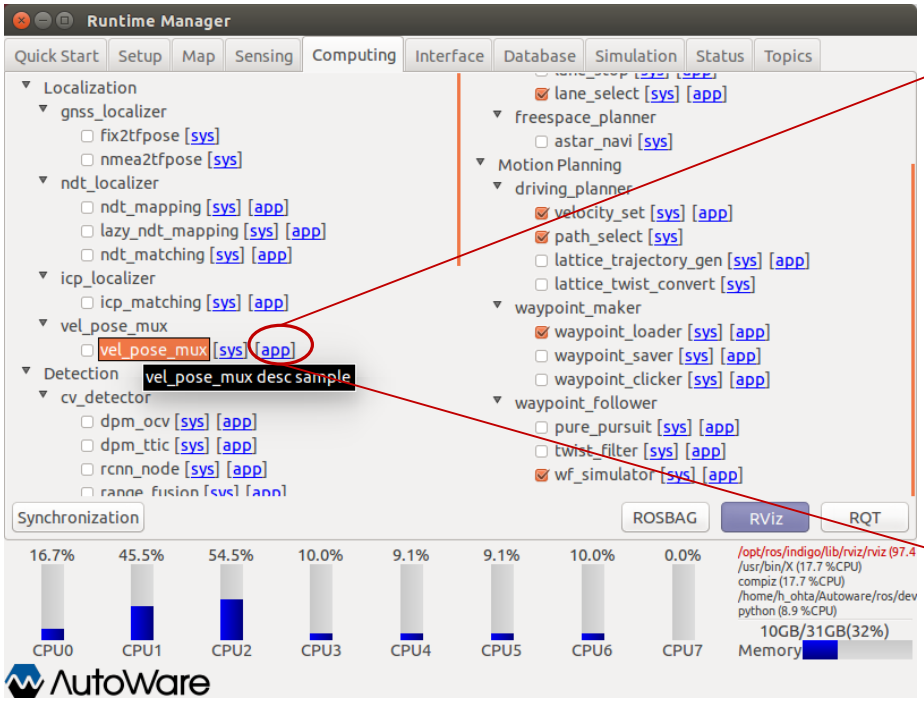
1. Specify the initial position by 2D Pose Estimate in Rviz
2. A vehicle model will be displayed by step 1
   If the position of the model is not correct, repeat step 1. again

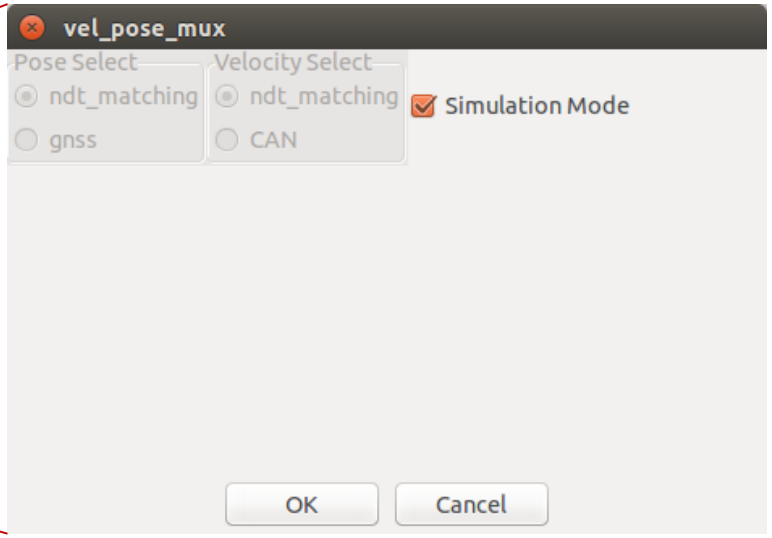


Path Following and Vehicle Control – Steps of wf_simulator (4/6)

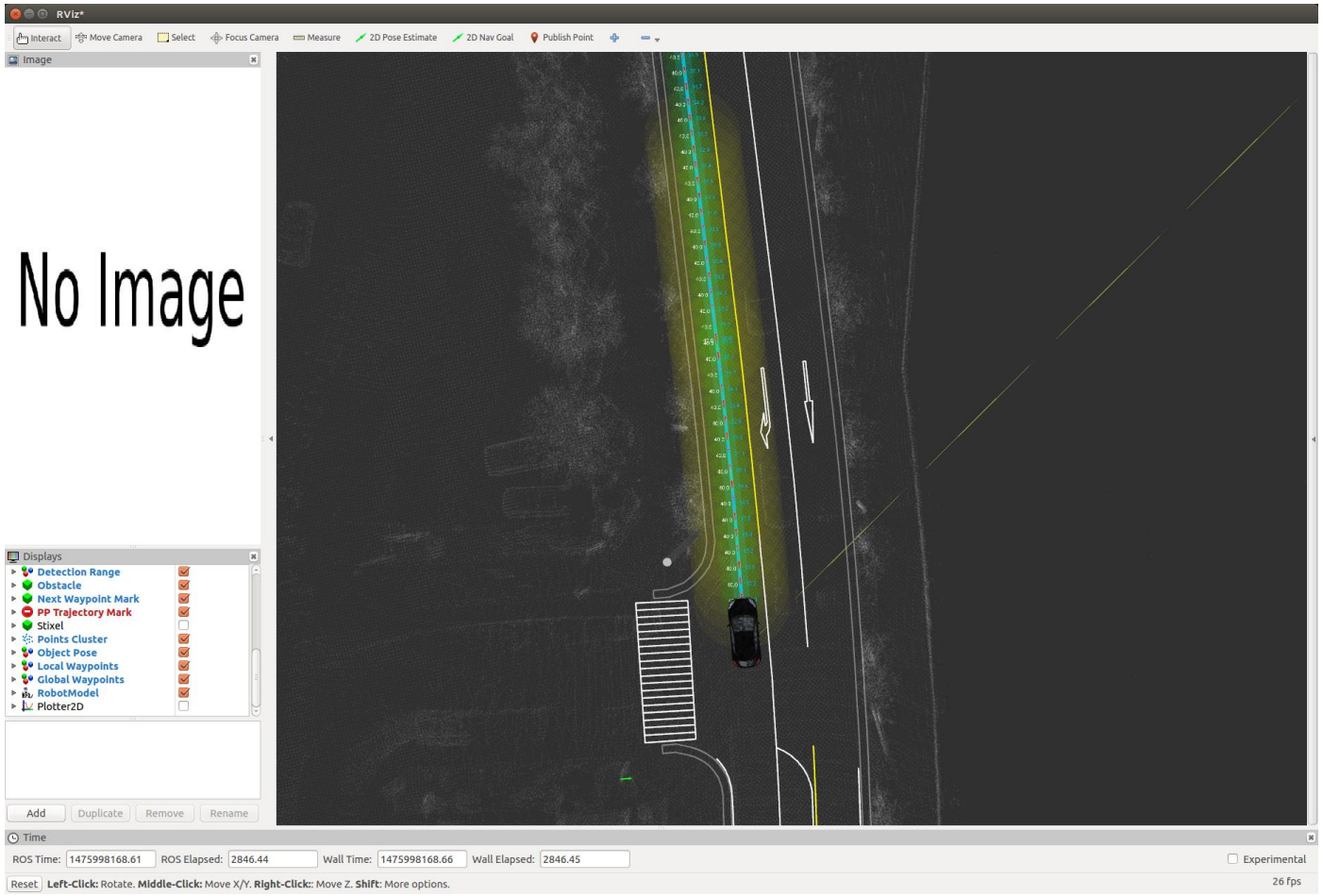
- Specifying a Path

1. Launch [velocity_set] and [path_select] by checking (☑) the boxes
2. Clicking [app] on [vel_pose_mux], the bottom-right figure will be displayed, and then check ☑ the box of Simulation Mode
3. Check (☑) the box of vel_pose_mux





Path Following and Vehicle Control – Steps of wf_simulator (5/6)

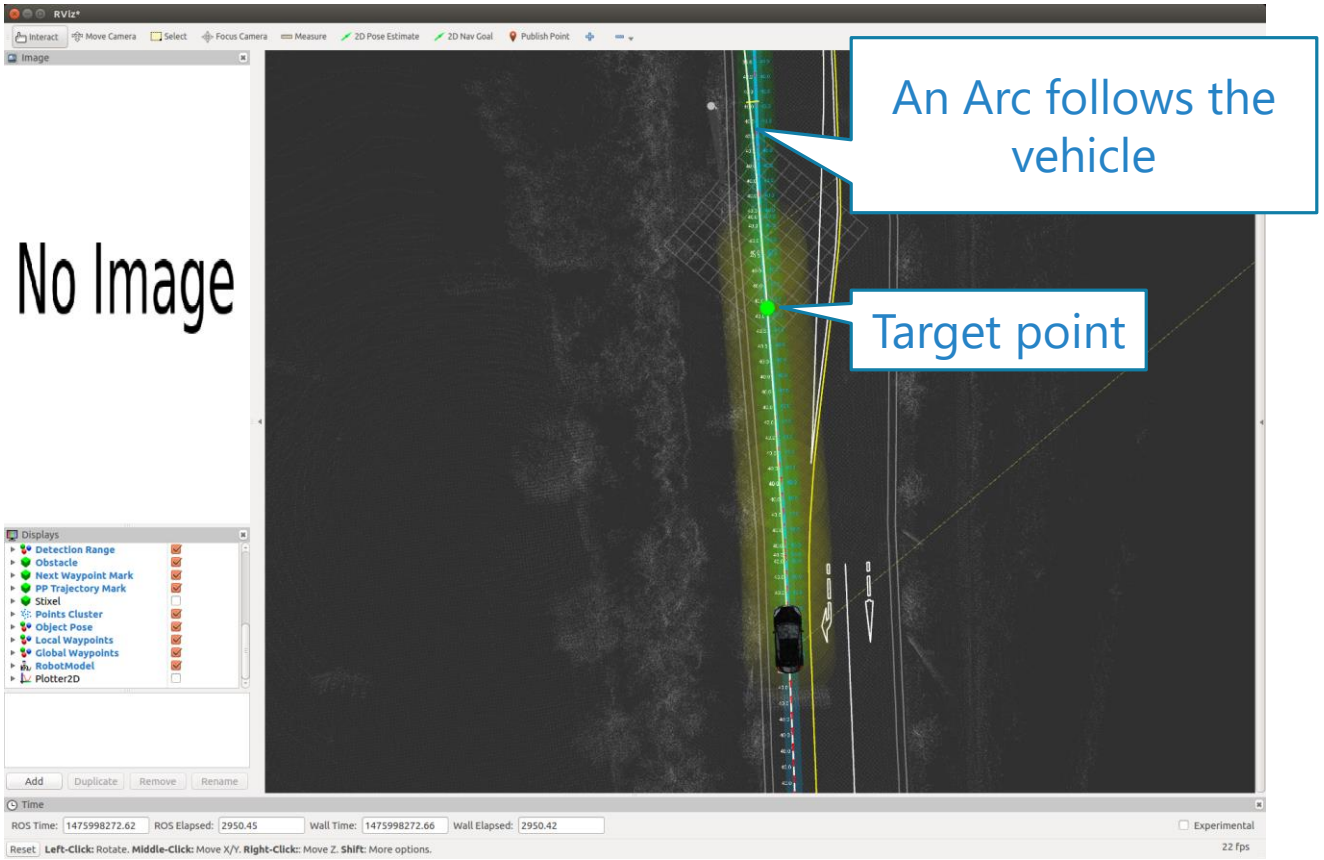


Path is shown as the above figure

Path Following and Vehicle Control – Steps of wf_simulator (6/6)

- Start path following

1. Launch [pure_pursuit] and [twist_filter]
2. Target points and the arc will be displayed, and then the simulator starts running



www.tier4.jp

Keeno
Consulting

http://www.keeno-consulting.com