

# Warning: Invalid frame ID "map" passed to canTransform argument source\_frame - frame does not exist | Vehicles in rviz are very far away from the map #5018

Answered by KYabuchi

XIAONNNN asked this question in Q&A



XIAONNNN on Jul 22

We tried to deploy our own sensors (lidar-rs16, gnss/imu-fixposition rtk2) and cars on the universe version. After following the tutorial to start autoware.universe, we found that only the point cloud map and not the car model is missing in rviz, the specific terminal error reporting command is as follows:

#### issue1:

[rviz2-86] Warning: Invalid frame ID "map" passed to canTransform argument target\_frame - frame does not exist

[rviz2-86] at line 156 in /tmp/binarydeb/ros-galactic-tf2-0.17.5/src/buffer\_core.cpp

[multi\_object\_tracker-57] Warning: Invalid frame ID "map" passed to canTransform argument source\_frame - frame does not exist

[multi\_object\_tracker-57] at line 156 in /tmp/binarydeb/ros-galactic-tf2-0.17.5/src/buffer\_core.cpp

[motion\_velocity\_smoothen-66] [WARN] [1721641469.672513022] [planning.scenario\_planning.motion\_velocity\_smoothen]: failed to get transform from map to base\_link: "map" passed to lookupTransform argument target\_frame does not exist.

#### issue2:

[component\_container-27] [INFO] [1721641470.288783916]

[map.vector\_map\_tf\_generator]: broadcast static tf. map\_frame:map, viewer\_frame:viewer, x:42420.7, y:71757.8, z:0

[gyro\_odometer-43] [WARN] [1721641470.368633513]

[localization.twist\_estimator.gyro\_odometer]: Imu msg is not subscribed

[component\_container\_mt-79] [WARN] [1721641470.444201457]

[autoware\_api.external.vehicle\_status]: The turn\_indicators topic is not subscribed

[ekf\_localizer-44] [WARN] [1721641470.551851287]

[localization.pose\_twist\_fusion\_filter.ekf\_localizer]: The node is not activated. Provide initial pose to pose\_initializer

[multi\_object\_tracker-57] [WARN] [1721641470.627448320]

[multi\_object\_tracker]: Could not find a connection between 'base\_link' and 'map' because they are not part of the same tree.Tf has two or more unconnected trees.

#### issue3:

[component\_container\_mt-67] [INFO] [1721641467.230712893]

[planning.scenario\_planning.lane\_driving.behavior\_planning.behavior\_path\_planner]: waiting for scenario\_topic

#### Category



Q&A

#### Labels

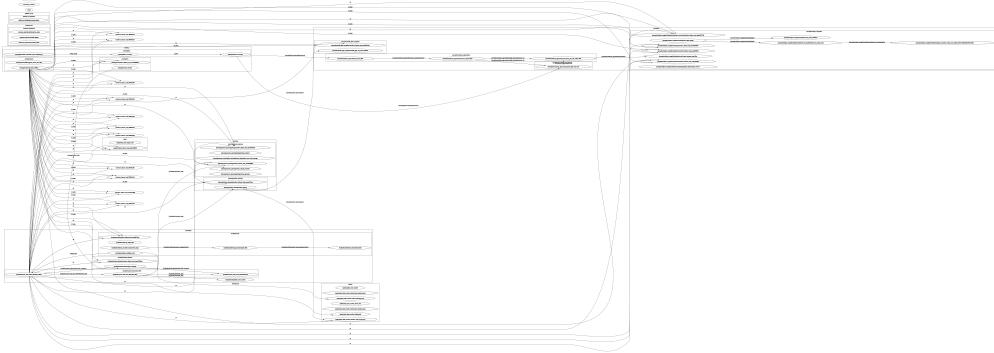
component:localiza...

#### 4 participants



Here we attach the corresponding log, tf-tree and rqt\_grah:

[launch.log](#)  
[frames.pdf](#)



Hope someone can answer our doubts.

↑ 1

✓ Answered by **KYabuuchi** on Aug 12

@silly-h /localization/twist\_estimator/twist\_with\_covariance represents velocity and angular velocity, so it should be close to zero when the vehicle is stationary.

The *linear.x* component of this topic is assigned the value of *linear.x* from vehicle\_velocity\_converter/twist\_with\_covariance and angular.z is

[View full answer ↓](#)

3 comments · 16 replies

Oldest

Newest

Top



KYabuuchi on Jul 23

Collaborator

Hi. @XIAONNNN

The reason your car model isn't showing in RViz and issues 1-3 are occurring is because the `map->base_link` TF is not being estimated.

Autoware requires the initial position estimation to be completed to estimate the map->base\_link TF. Looking at the launch.log, it seems that the pose\_initializer is attempting and failing to estimate the initial position periodically.

If the point cloud map is being displayed, refer to [this page](#) and try the 2D Pose Estimate tool. This will initiate the initial position estimation, and the vehicle model should then be rendered.

Autoware also provides automatic initial position estimation using GNSS. From the launch.log, it seems that GNSS is not available in your environment. If this is not intentional, you should check the GNSS topic to ensure it is functioning correctly.

↑ 2

15 replies

Show 11 previous replies

silly-h on Aug 5

@KYabuuchi Please refer to the attached TF tree for our TF configuration. Regarding the TF tree issue, we have also found that there is a TF transformation from `base_link` to `map` in the `/TF`, but its publishing frequency is very low, which may be why we did not see the information from `base_link` to `map` when using the `ros2 run tf2_tools view_frames` command.

```
...  
transforms:  
- header:  
  stamp:  
    sec: 1722591680  
    nanosec: 512326485  
    frame_id: map  
  child_frame_id: base_link  
  transform:  
    translation:  
      x: -2.460859309278365  
      y: 3.6561691919307595  
      z: -0.8143382619465525  
    rotation:  
      x: 0.006557707623514636  
      y: 0.014581543689070477  
      z: 0.9148991869288473  
      w: 0.4033656564598707  
...  
transforms:  
- header:  
  stamp:  
    sec: 1722591680  
    nanosec: 514730096  
    frame_id: odom  
  child_frame_id: base_link  
  transform:  
    translation:  
      x: 0.0  
      y: 0.0  
      z: 0.0  
    rotation:  
      x: 0.0
```

[frames.pdf](#)

Additionally, regarding the timestamp issue, we can try to reduce the time difference between the `/clock`, `/sensing/vehicle_velocity`, and `/sensing imu/imu_data` topics. However, we are puzzled as to why there is a time difference of around 3 seconds between the `/sensing/pose_estimator` and `/localization/util/downsample` topics and our sensor topics. Could this be related to the synchronization of our sensor timestamps?

KYabuuchi on Aug 5 Collaborator

@silly-h Have you modified the `pose_frame_id` parameter in the [`ekf\_localizer.param.yaml`](#) from `map` to `odom` ?

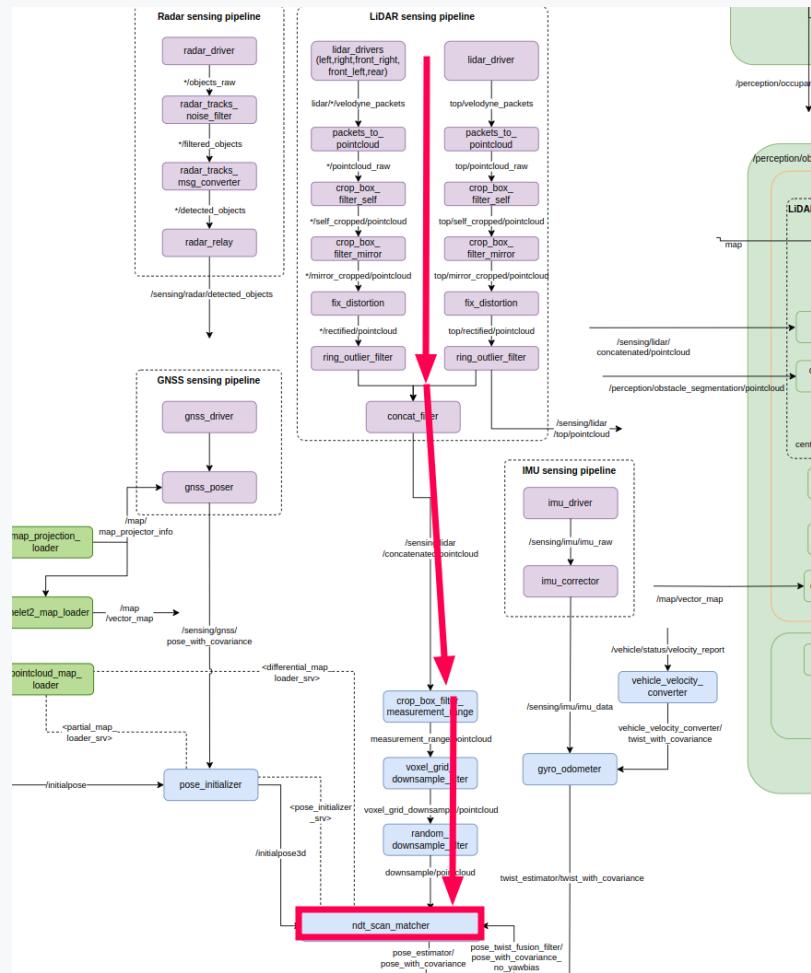
This parameter should be set to `map`. Also, Autoware does not use the `odom` `frame_id`.

Normally, the TF between `map` and `base_link` should be published at a frequency of 50Hz by `ekf_localizer`.

But in your frames.pdf diagram, it seems that the transformation between `odom` and `base_link` is being published at 50Hz. 😕

`/localization/util/downsample` is relayed by many nodes, and point cloud preprocessing can be time-consuming. To investigate the root cause, it would be effective to refer the node diagram and identify where the latency is occurring.

If the delay is introduced at the lidar\_packets stage, there may be an issue with sensor synchronization.



silly-h on Aug 12

@KYabuuchi Thank you very much for your reminder. We have already attempted to reduce the sensor time differences and confirmed that the TF settings between `map` and `base_link` are correct.

Additionally, we checked our `/localization/twist_estimator/twist_with_covariance` topic and noticed that the output remains constant.

```
---
```

```
header:  
  stamp:  
    sec: 1722850417  
    nanosec: 500215112  
  frame_id: base_link  
twist:  
  twst:  
    linear:  
      x: 0.0  
      y: 0.0  
      z: 0.0  
    angular:  
      x: 0.0  
      y: 0.0  
      z: 0.0  
  covariance:  
  - 0.04000000000000001  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 10000.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 10000.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0009  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0009  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0  
  - 0.0009
```

```
--
```

Is this value supposed to be stable when the vehicle is stationary? If it's not normal, could it affect the publishing of the `ekf_localizer` node?

The following two images show the `/sensing/imu/imu_data` and `vehicle_velocity_converter/twist_with_covariance` topics from top to bottom, respectively. Once again, thank you very much for your help.

```
---
```

```
header:  
  stamp:  
    sec: 1722591720  
    nanosec: 348112000  
    frame_id: base_link  
orientation:  
  x: 0.0  
  y: 0.0  
  z: 0.0  
  w: 1.0  
orientation_covariance:  
- 0.0  
- 0.0  
- 0.0  
- 0.0  
- 0.0  
- 0.0  
- 0.0  
- 0.0  
angular_velocity:  
  x: 0.00237205710019366  
  y: 0.0025978278685098105  
  z: -0.0005916219054553477  
angular_velocity_covariance:  
- 0.0009  
- 0.0  
- 0.0  
- 0.0  
- 0.0009  
- 0.0  
- 0.0  
- 0.0  
- 0.0009  
linear_acceleration:  
  x: 0.0709529369407396  
  y: 0.035820425410340084  
  z: 9.80581245344132  
linear_acceleration_covariance:  
- 100000000.0  
- 0.0  
- 0.0  
- 0.0  
- 100000000.0  
- 0.0  
- 0.0  
- 0.0  
- 100000000.0  
---
```

```
header:
  stamp:
    sec: 1722591709
    nanosec: 66234350
  frame_id: base_link
twist:
  twist:
    linear:
      x: 0.0
      y: 0.0
      z: 0.0
    angular:
      x: 0.0
      y: 0.0
      z: 0.0
  covariance:
  - 0.04000000000000001
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 10000.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.0
  - 0.01000000000000002
```



KYabuuchi on Aug 12 Collaborator

@silly-h `/localization/twist_estimator/twist_with_covariance` represents velocity and angular velocity, so it should be close to zero when the vehicle is stationary.

The `linear.x` component of this topic is assigned the value of `linear.x` from `vehicle_velocity_converter/twist_with_covariance`, and `angular.z` is assigned the value of `angular.z` from `/sensing imu imu_data`.

Additionally, if both values are very small, `twist_estimator` sets the angular velocity to zero before publishing. This is to prevent the estimated position from drifting slowly over time due to the IMU yaw bias when the vehicle is not moving.

Please refer to [source code](#) for details.

If you suspect

/localization/twist\_estimator/twist\_with\_covariance , it would be a good idea to verify that linear.x is non-zero when the vehicle is moving and angular.z is non-zero when the vehicle is turning.

 Marked as answer

Answer selected by liuXinGangChina



liuXinGangChina on Jul 29 Collaborator

the tf that yabuchi-san mentioned above "map to base-link", usually given by ekf and ekf use ndt for input.

make sure your ndt works firtly

then check whether ekf functional

for beginer ,forget about using gnss initializaiton. you can use rviz init pose plugin to drag the pose manually

happy hacking!

 2

 2

0 replies



liuXinGangChina on Aug 7 Collaborator

the log.rar is not accessable now

like yabuchi-san said you should use a 3d map for 3d localization, did you delete the z info of the map by mistake? accually most pcd file generated by lidar slam is 3d

 2

 2

1 reply



silly-h on Aug 12

@liuXinGangChina Thank you very much for your reminder. We mistakenly set the Z value of the point cloud to 0 when processing the point cloud map, which led to errors in NDT matching calculations. We have now resolved this issue and completed the pose initialization.