

ROS可視化最先端

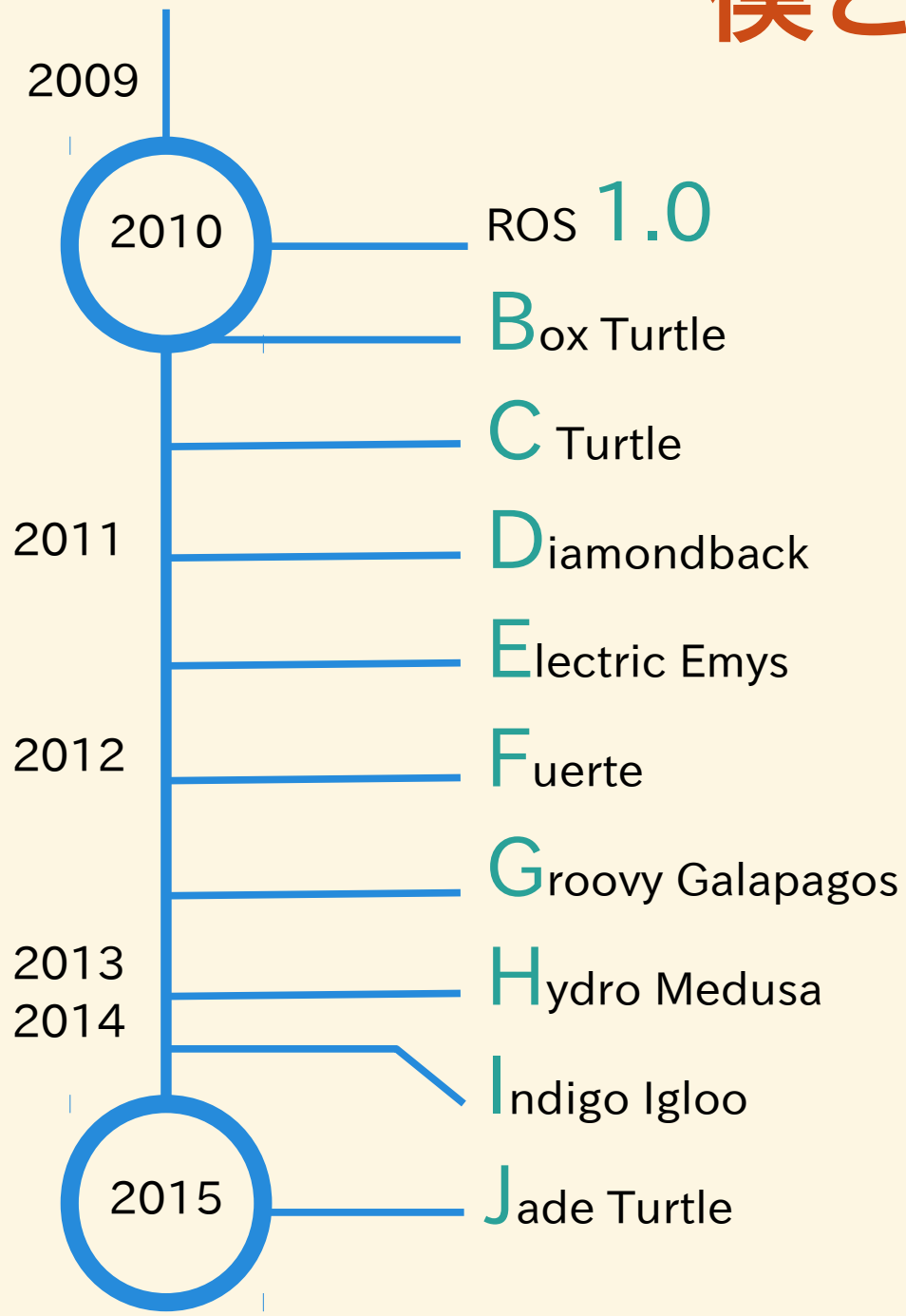
自己紹介

- 名前
 - 植田亮平 (@graemon_coder)
- 所属
 - 東京大学情報理工学系研究科・創造情報学専攻
 - JSK/情報システム工学研究室 (稲葉・岡田研)
- 専門
 - ロボットシステム
 - システム構成、動作計画、画像処理、点群処理
- 就職先募集中

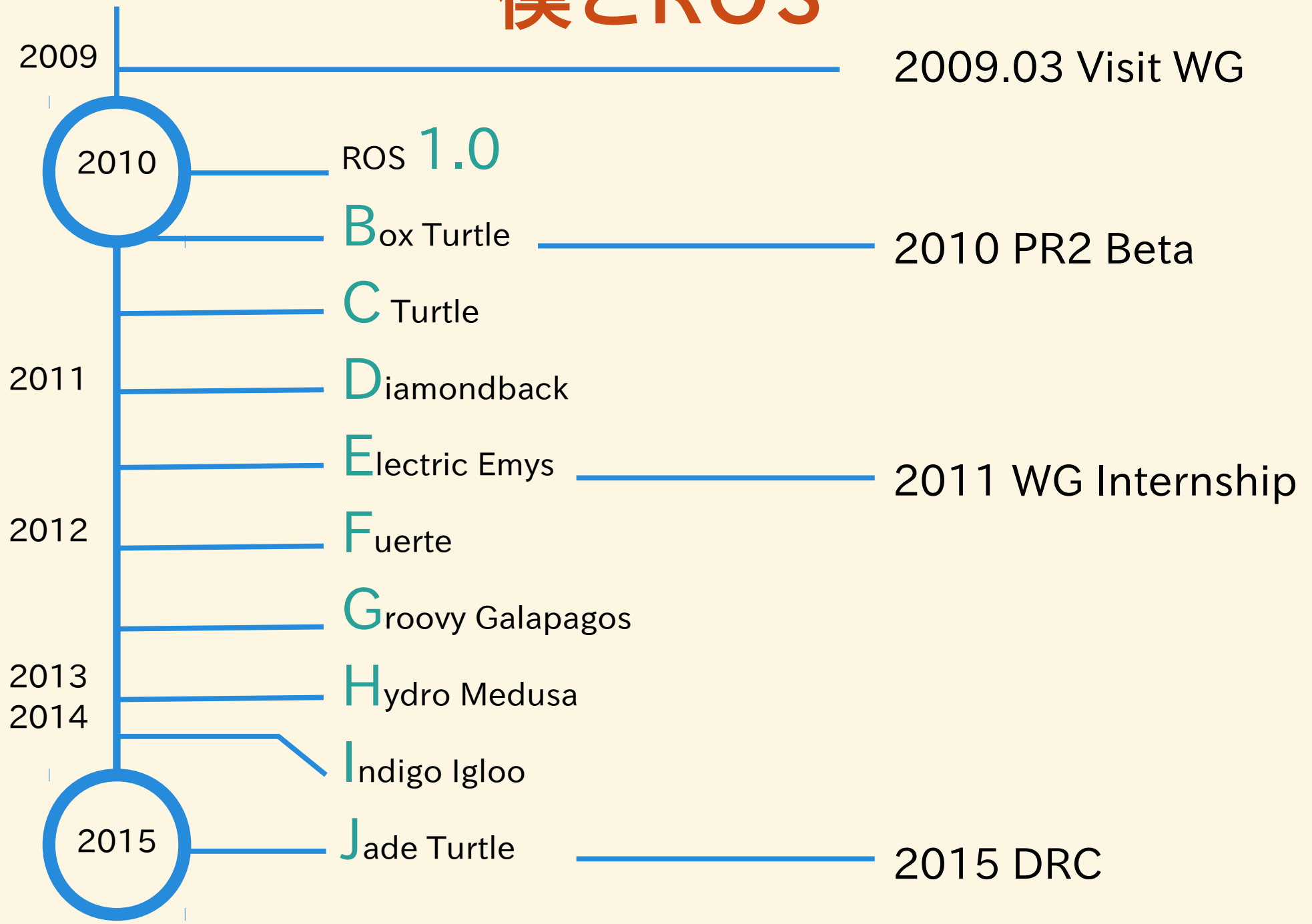
僕とROS



僕とROS



僕とROS



Quick Introduction of jsk-ros-pkg

- **jsk-ros-pkg** [<https://github.com/jsk-ros-pkg>]
 - 稲葉・岡田研のレポジトリ
 - Over 100 packages
- **Highlights**
 - jsk_common/jsk_topic_tools
 - C++ Utility Library for ROS
 - jsk_roseus/roseus
 - Euslisp binding for ROS
 - jsk_recognition/jsk_pcl_ros
 - PCL and OpenCV based nodelets
 - [jsk_visualization/jsk_rviz_plugins](#)
 - Miscellaneous rviz plugins
 - [jsk_visualization/jsk_rqt_plugins](#)
 - Miscellaneous rqt plugins

今日の話

- 可視化について話します

可視化とRobotics

- 可視化は最も重要と言っても過言ではない
 - デバッグ
 - システム構成

ROS = Build System

+

Communication Library

+

Community

catkin, rosdep, rospack



ROS = Build System

+

Communication Library

+

Community

ROS = Build System

+

Communication Library

+

Community

 Github, answers.ros.org

ROS = Build System

+

Communication Library

+

Community

→ Topic, Service, Parameter

ROS = Build System

+

Communication Library

+

Topic, Service, Parameter

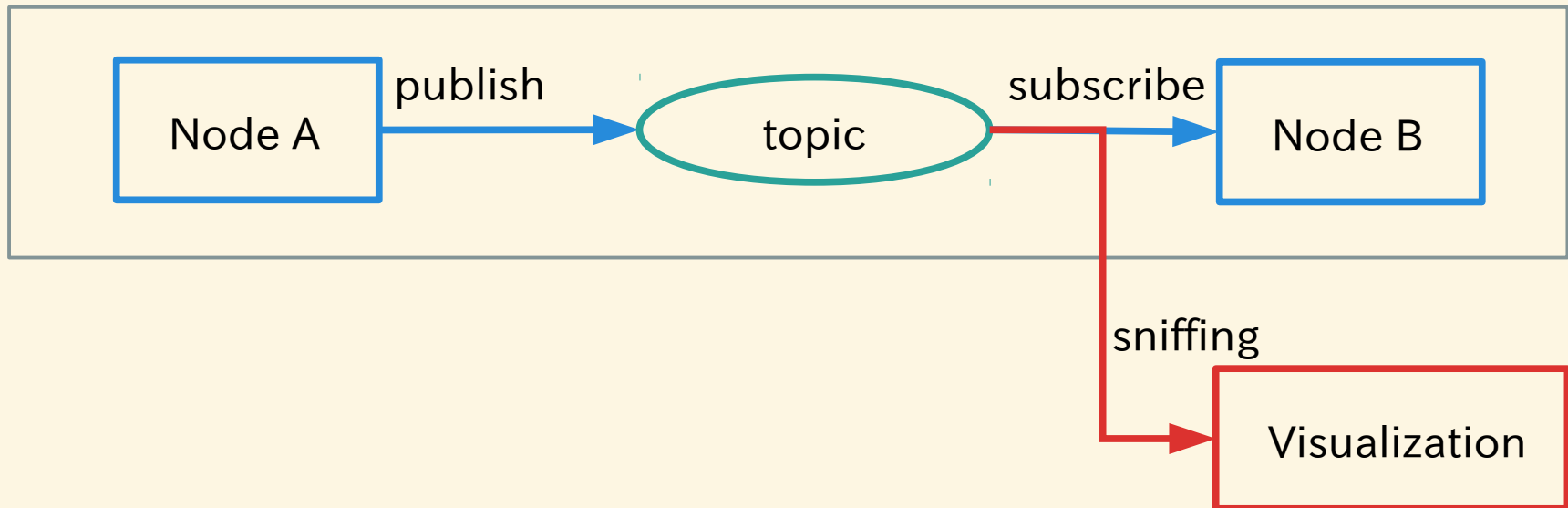
Community

Pub/Subモデルの通信が可視化によってすごく恩恵が大きかった
メッセージの型 → 可視化の対応

Visualization: Why ROS Awesome?

- 「行儀の良い」可視化は難しい
 - 可視化によってメイン処理が遅くなる
 - GUI絡みのマルチスレッド化の難しさ
- ROSの通信ライブラリの恩恵
 - メイン処理と可視化プロセスの分離が容易に

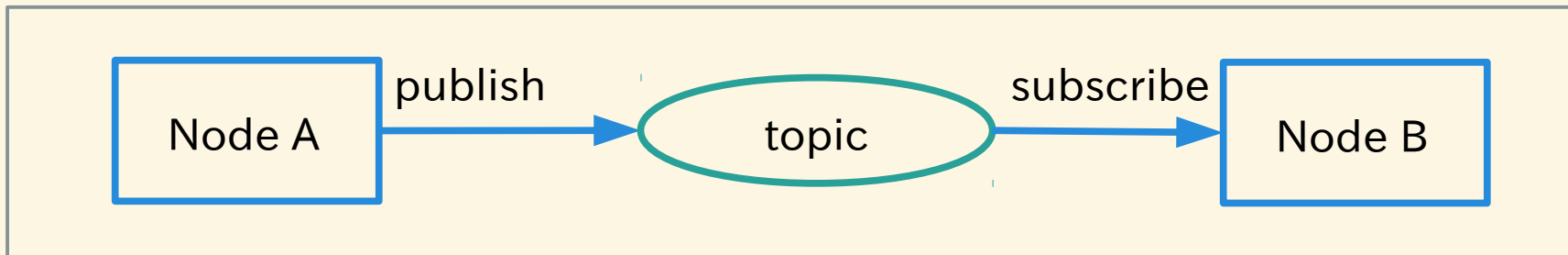
Main Process



Visualization: Why ROS Awesome?

- 「行儀の良い」可視化は難しい
 - 可視化によってメイン処理が遅くなる
 - GUI絡みのマルチスレッド化の難しさ
- ROSの通信ライブラリの恩恵
 - メイン処理と可視化プロセスの分離が容易に

Main Process

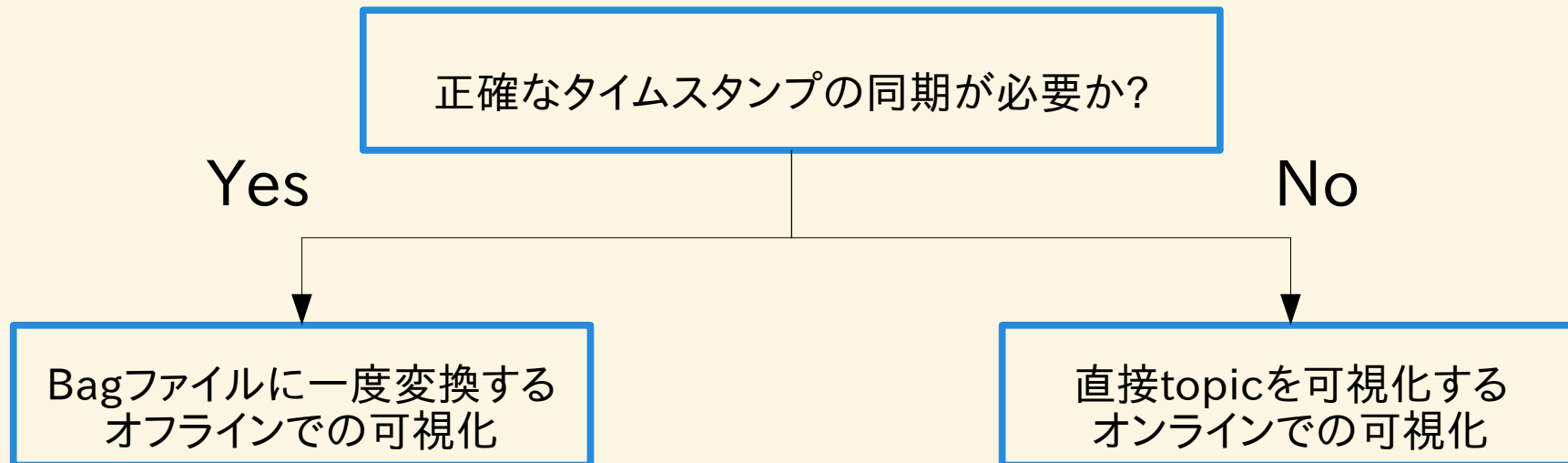


ROS Visualization Tools

- **rviz**
 - 3-D Visualization
- **rqt**
 - 2-D Visualization

Which tool?

- どの可視化ツールを使うべきか、作るべきかは自分が何を必要としているかに依存する



rqt?

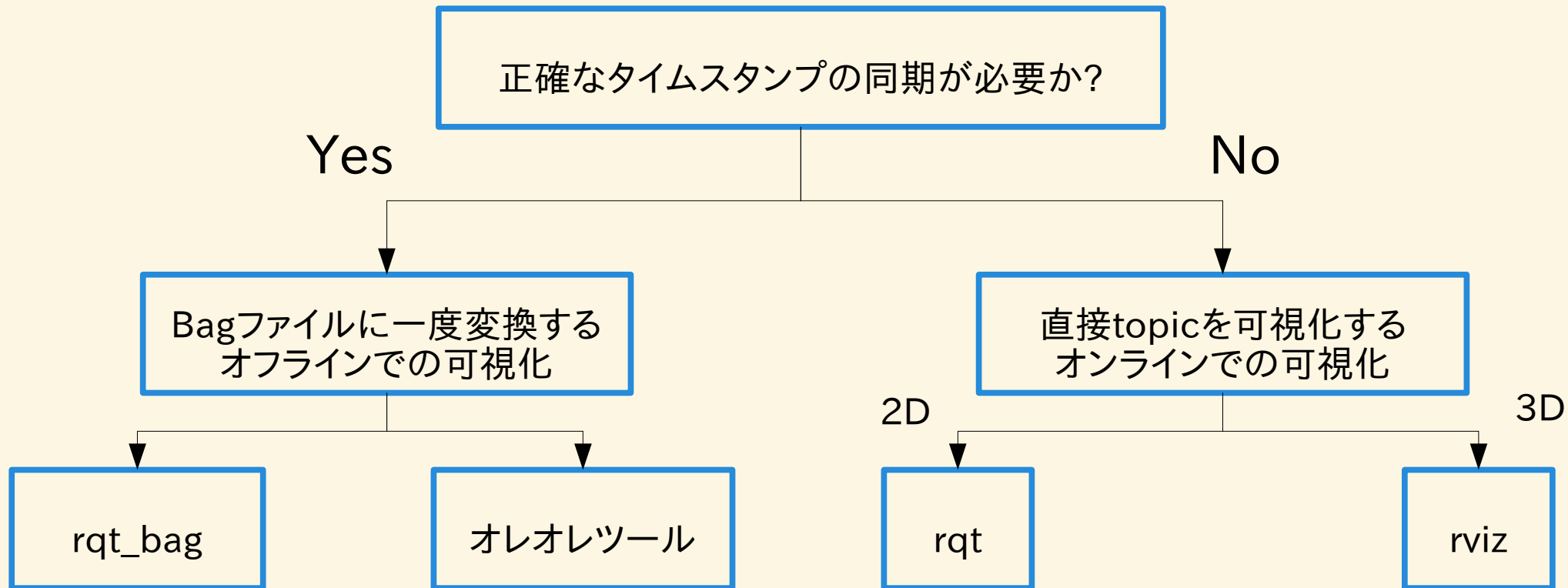
- Qtをベースとした可視化ツールセット
- プラグインシステム (オレオレwidget作れる)
- 実装はpython or C++で行う

rviz?

- Ogreをベースとした可視化ツールセット
- プラグインシステム (オレオレwidget作れる)
- 実装はC++で行う

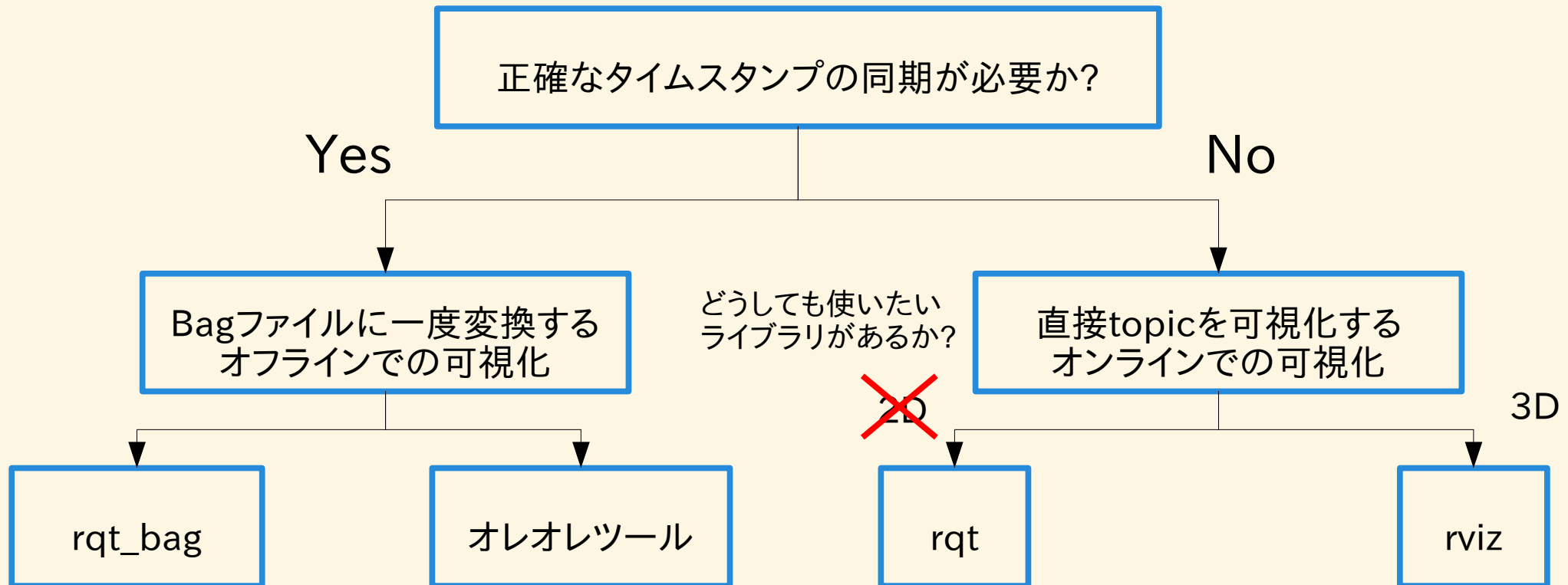
Which tool?

- どの可視化ツールを使うべきか、作るべきかは自分が何を必要としているかに依存する



Which tool for experts?

- どの可視化ツールを使うべきか、作るべきかは自分が何を必要としているかに依存する

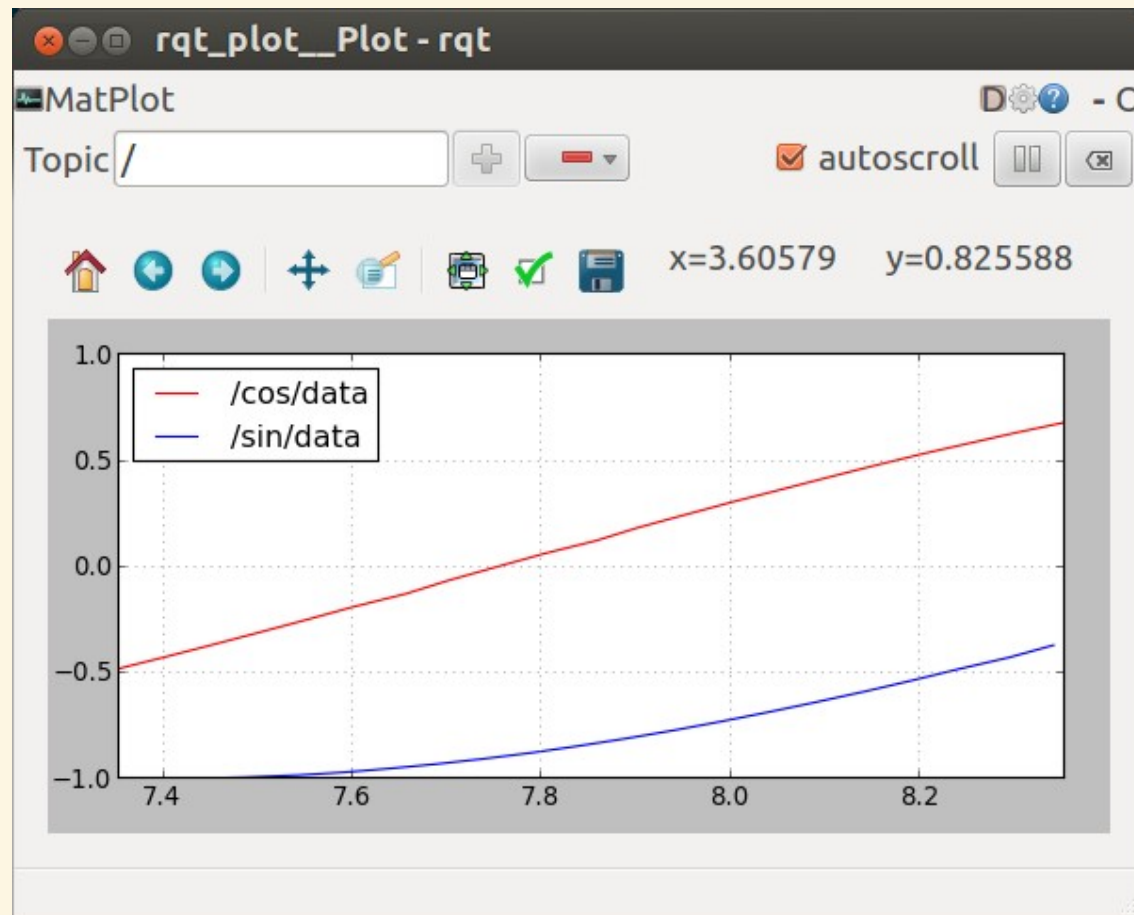


ROS official tools

- `rqt_plot`
- `rqt_bag`
- `image_view`
- `rviz`

rqt_plot

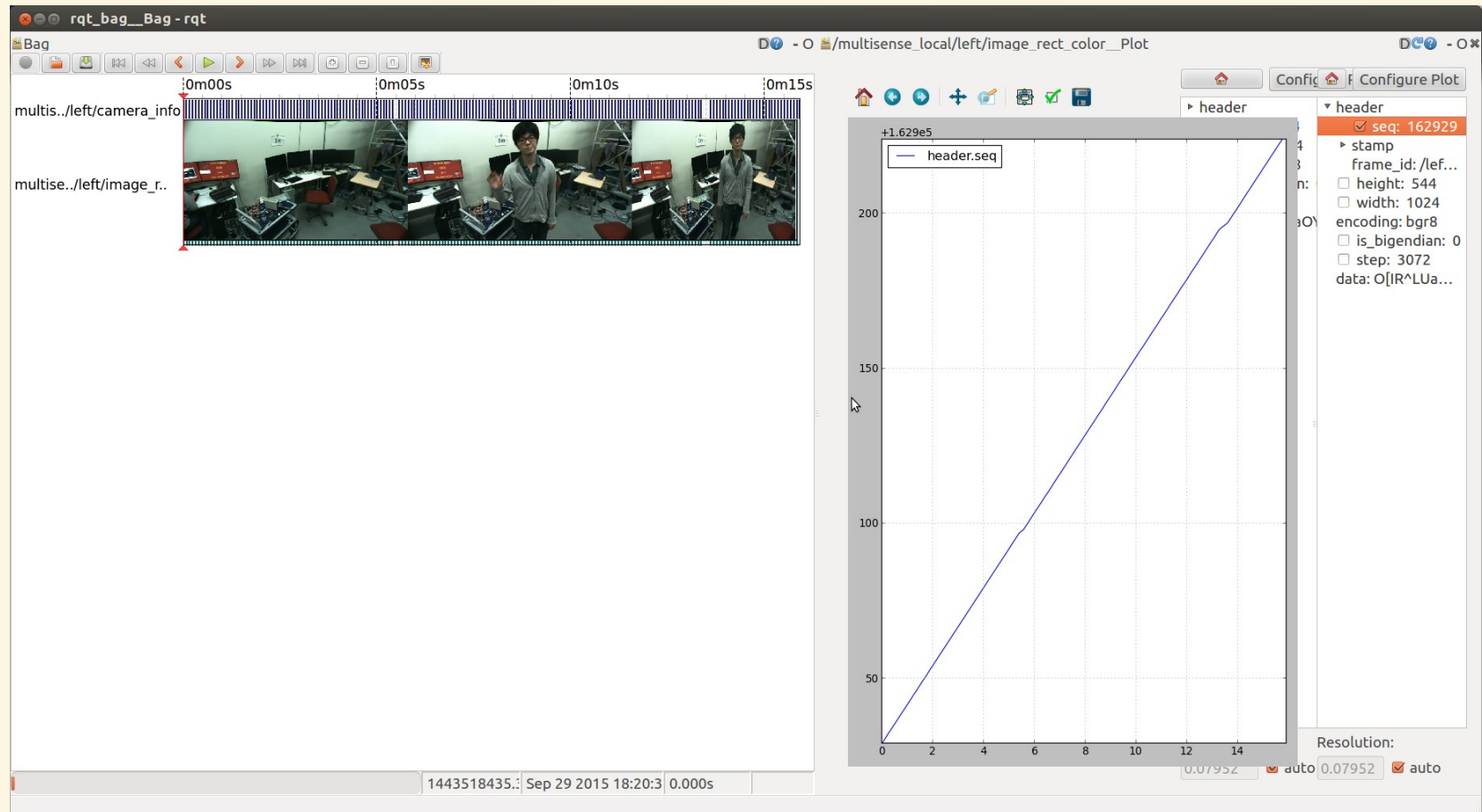
- 1次元数値データの可視化



→ t

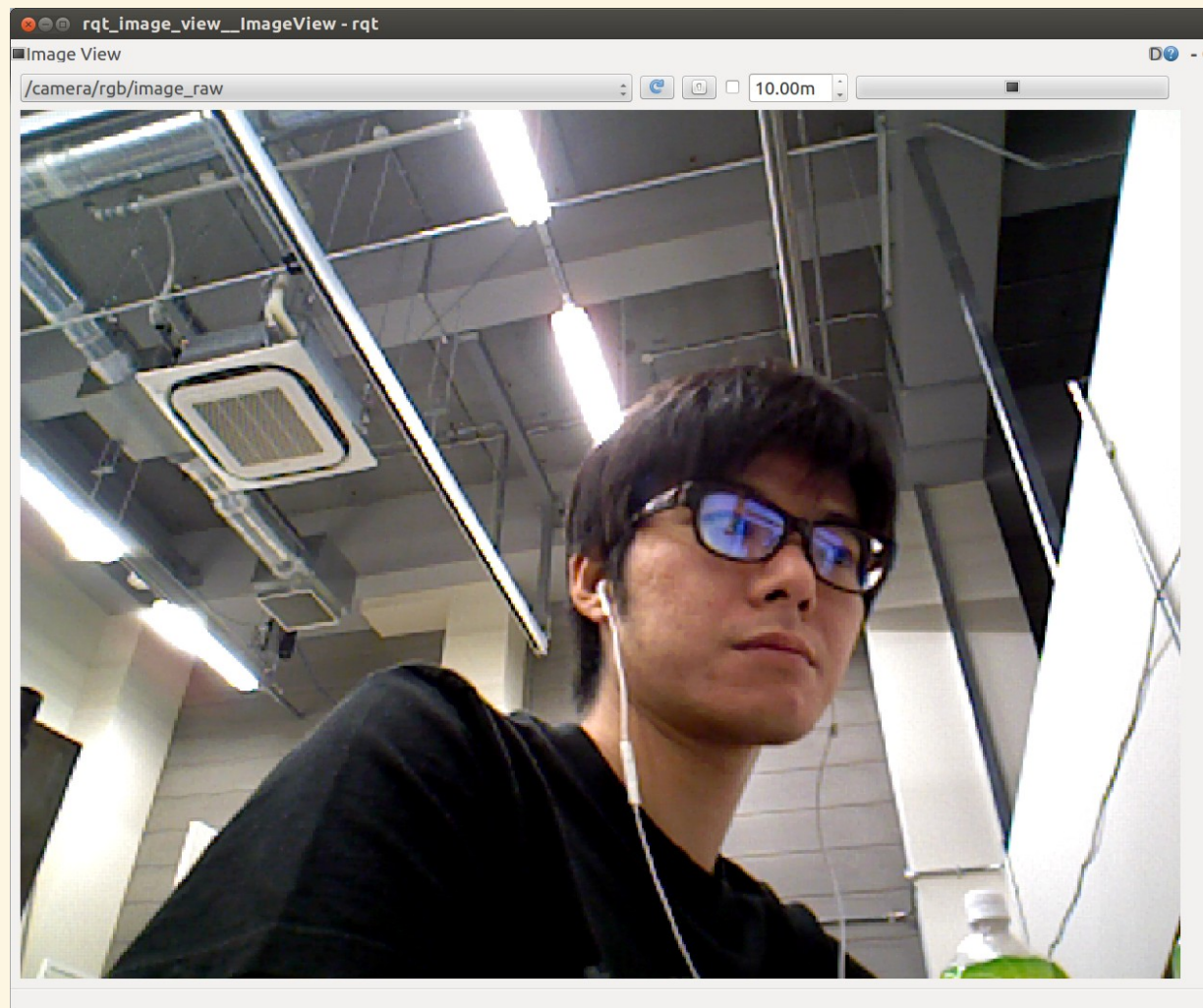
rqt_bag

- Bagファイルの可視化
 - プラグインを導入するとグラフのプロット、画像の表示などができる



image_view, rqt_image_view

- sensor_msgs/Imageの可視化



Visualization using jsk-ros-pkg

- 今日の本題
- Jsk-ros-pkgは良い可視化ツールを提供します
 - きっとあなたにも役立つものがこの中に
- Lineup
 - image_view2
 - bag_plotter.py
 - jsk_rqt_plugins
 - jsk_rviz_plugins

image_view2

- image_view + marker + UI
 - markerで色々描けるimage_view
 - ユーザの入力をとれるimage_view



jsk_tools/bag_plotter.py

- Bagファイルから直接複数のグラフを描画
 - rqt_bag的な感じ
- rqt_bagと違って
 - 複数のデータをひとつのグラフにかける
 - グラフのレイアウトを保存可能 (yamlで設定)

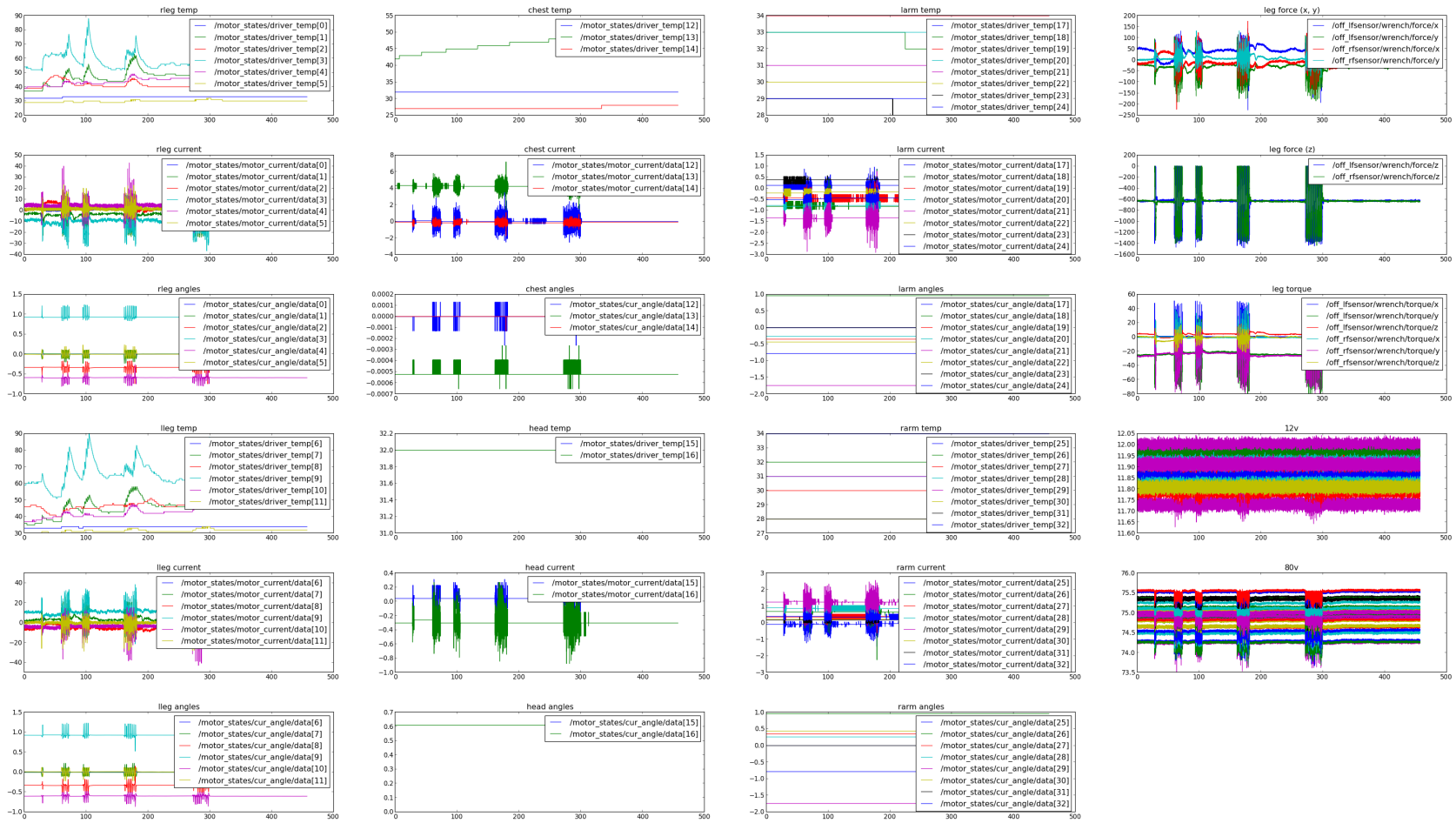
jsk_tools/bag_plotter.py

Figure 1

Figure 1

Plot from [Sat Aug 8 22:09:36 2015] to [Sat Aug 8 22:17:13 2015] (457 secs)

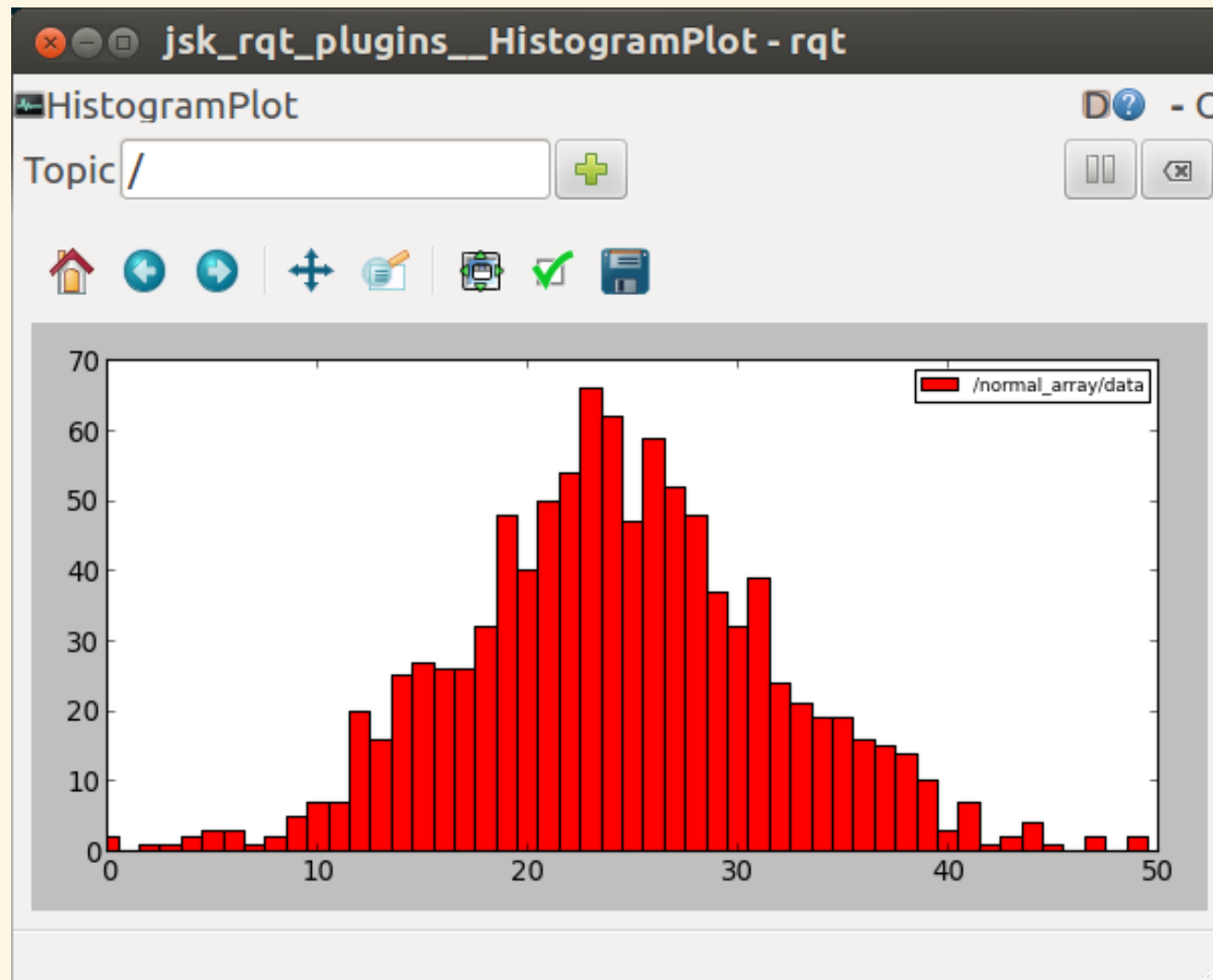
Removed /home/jeus/rosbags/2015/08/06/15/
roslab_2015-08-06-15-28-15_01.bag (104 MB)



toggle legend

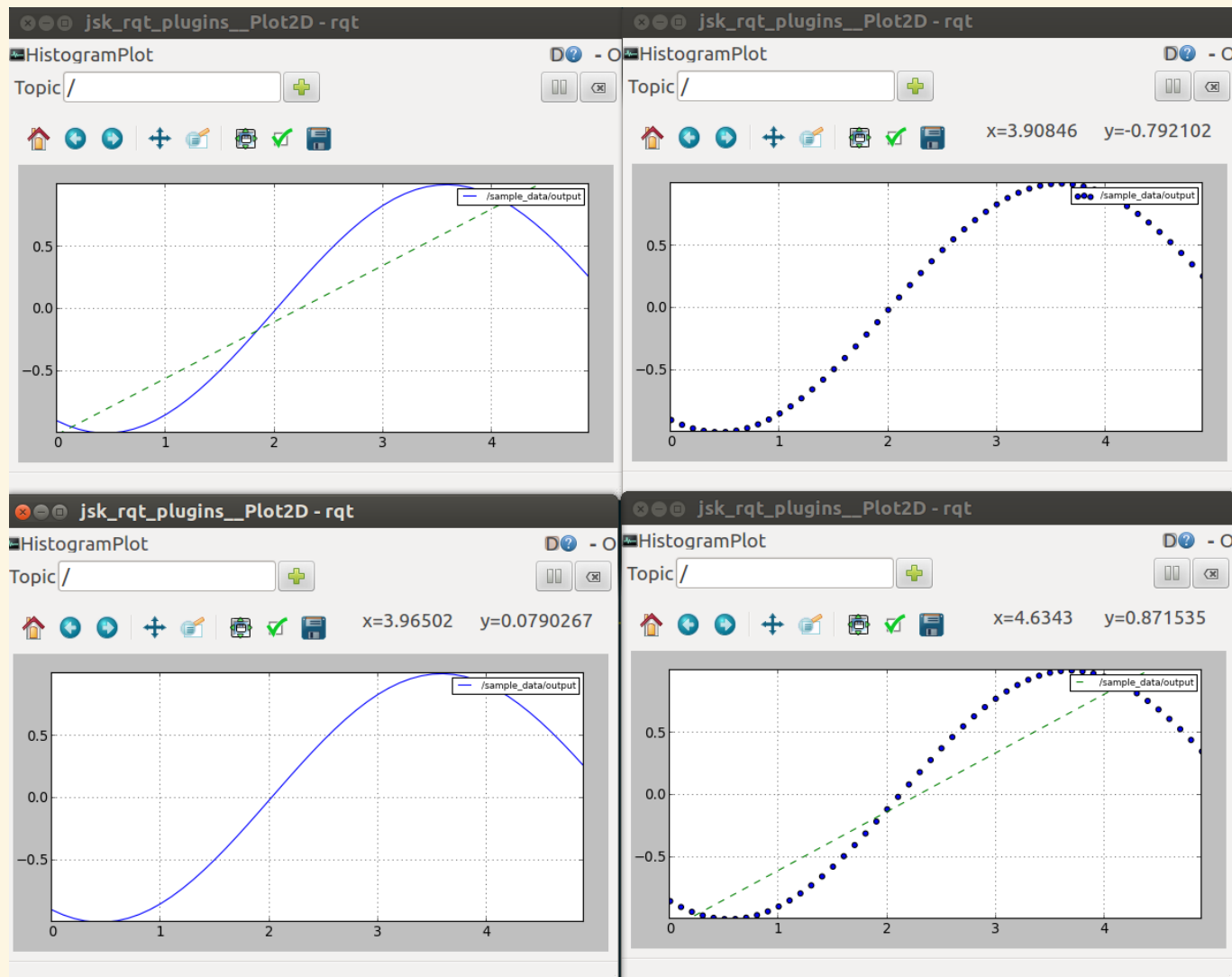
jsk_rqt_plugins rqt_histogram

- histogram可視化



jsk_rqt_plugins rqt_2dplot

- 2次元データ可視化 (rqt_plotは1次元データ)



jsk_rviz_plugins

- rvizの可能性を追求するパッケージ
 - 世界最先端のrvizパッケージを自負
- Ogre, rvizの良いサンプル

jsk_rviz_plugins PolygonArray + BoundingBox

- 複数の平面の可視化
- BoundingBoxの可視化
 - 認識で大活躍



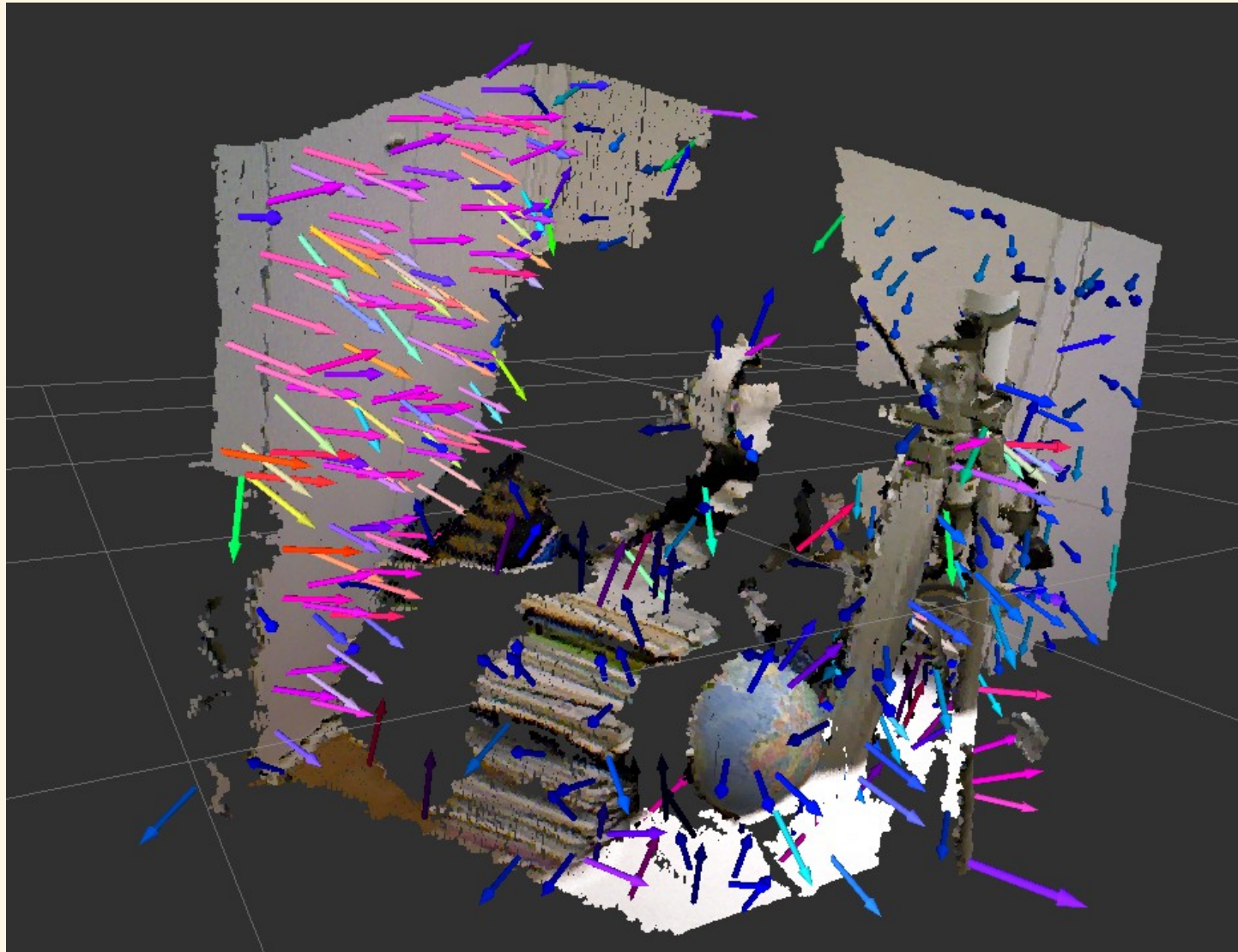
jsk_rviz_plugins CameraInfo

- カメラのIntrinsic Parameterの可視化



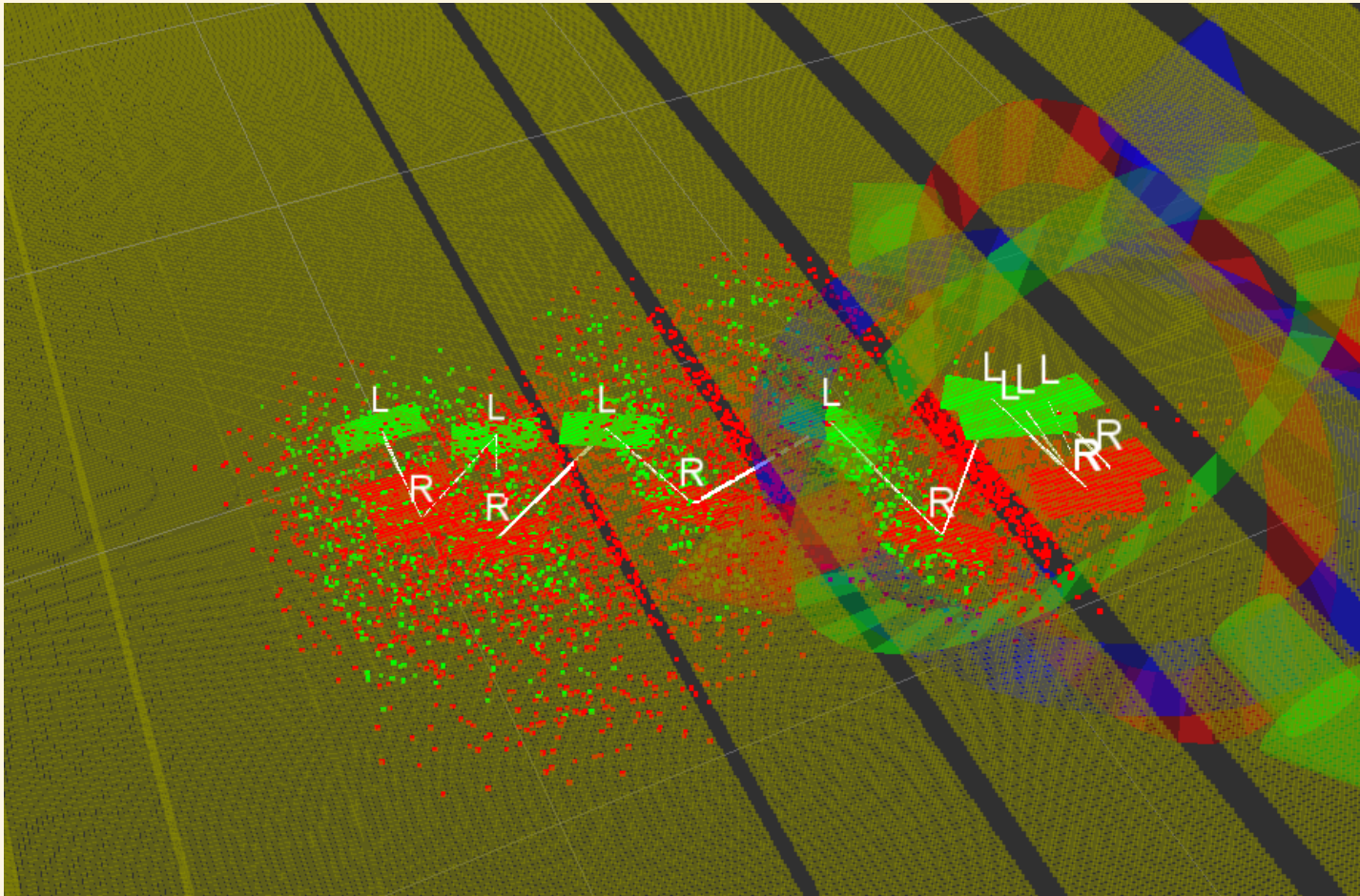
jsk_rviz_plugins Normal

- 法線ベクトルの可視化



jsk_rviz_plugins FootstepArray

- フットステップの可視化



jsk_rviz_plugins TFTrajectory

- TFの軌道の可視化



jsk_rviz_plugins TwistStamped

- TwistStampedの可視化



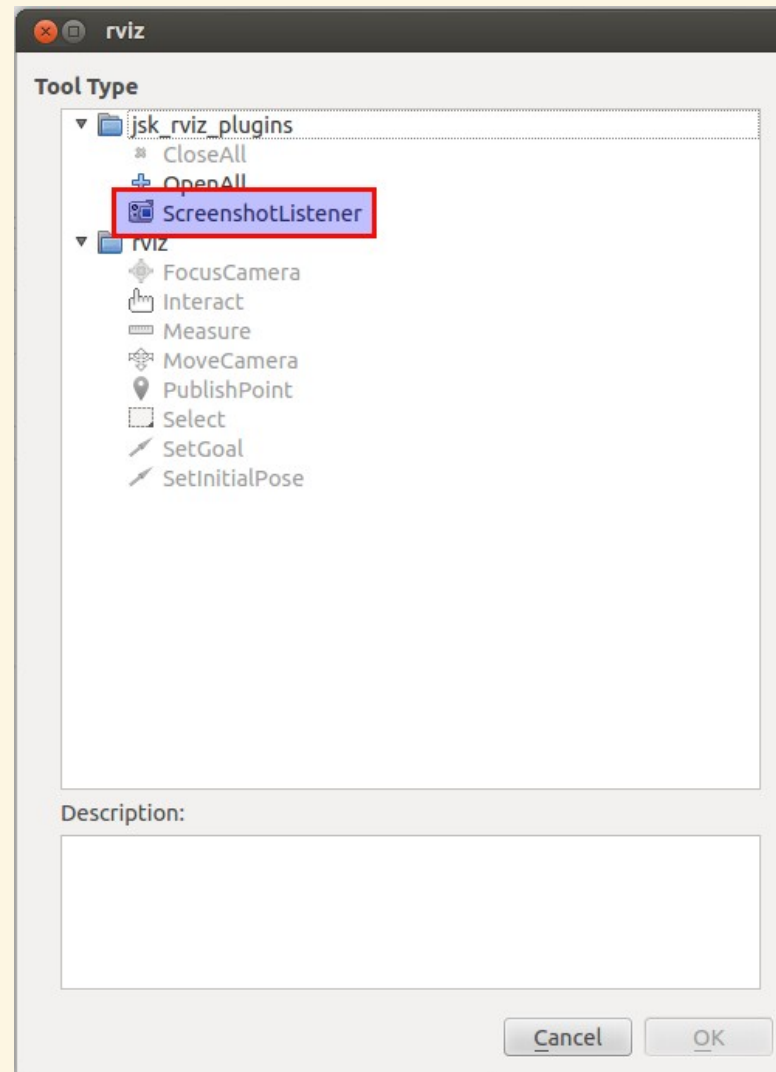
jsk_rviz_plugins VideoCapture

- 動画保存機能

  VideoCapture	<input checked="" type="checkbox"/>
start capture	
filename	output.avi
fps	30

jsk_rviz_plugins ScreenshotListener

- serviceでスクリーンショット



jsk_rviz_plugins PeoplePositionMeasurement Array

- 某殻機動隊風味の顔検出結果の可視化



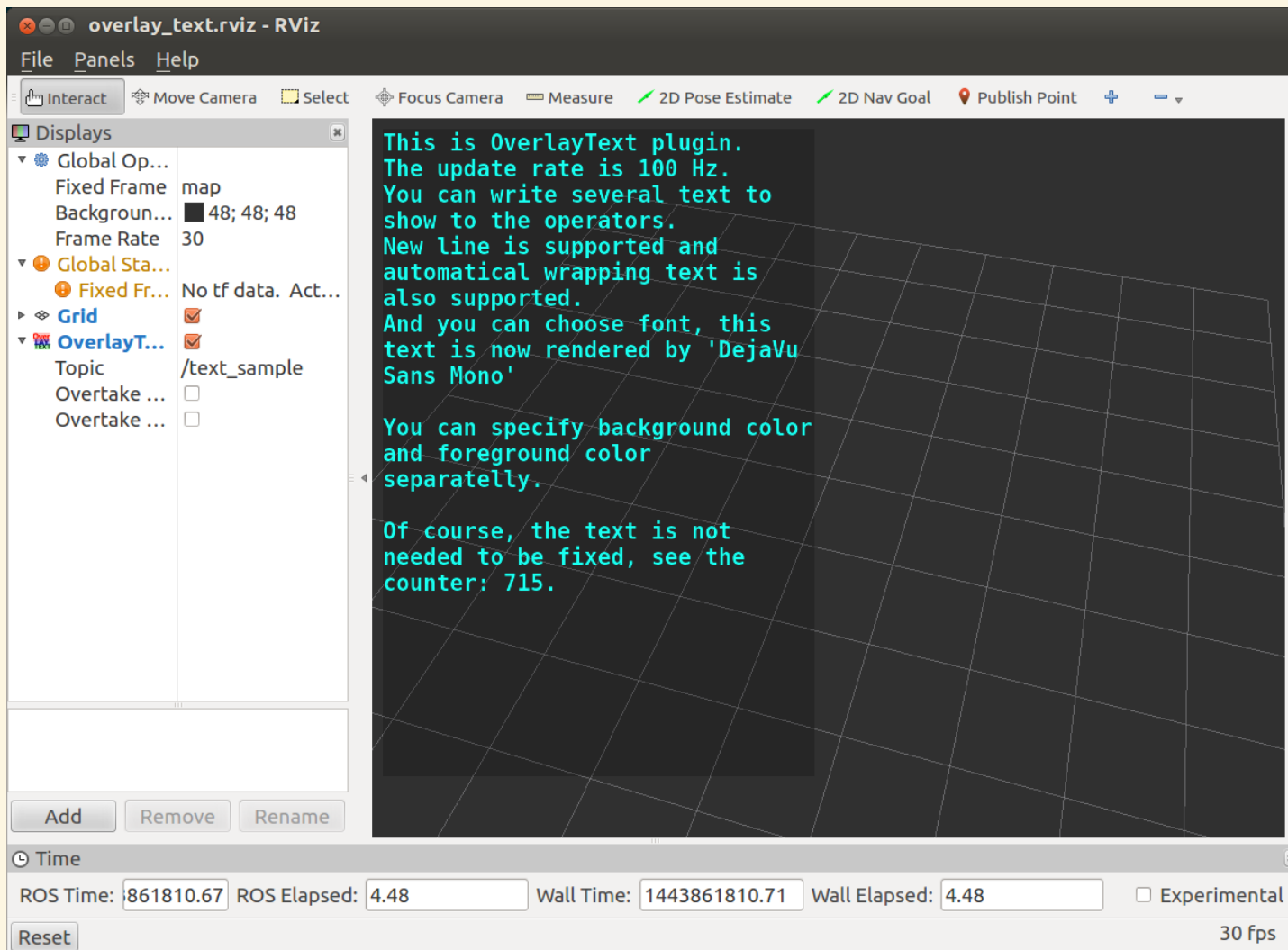
jsk_rviz_plugins Pictogram

- Pictogramの可視化



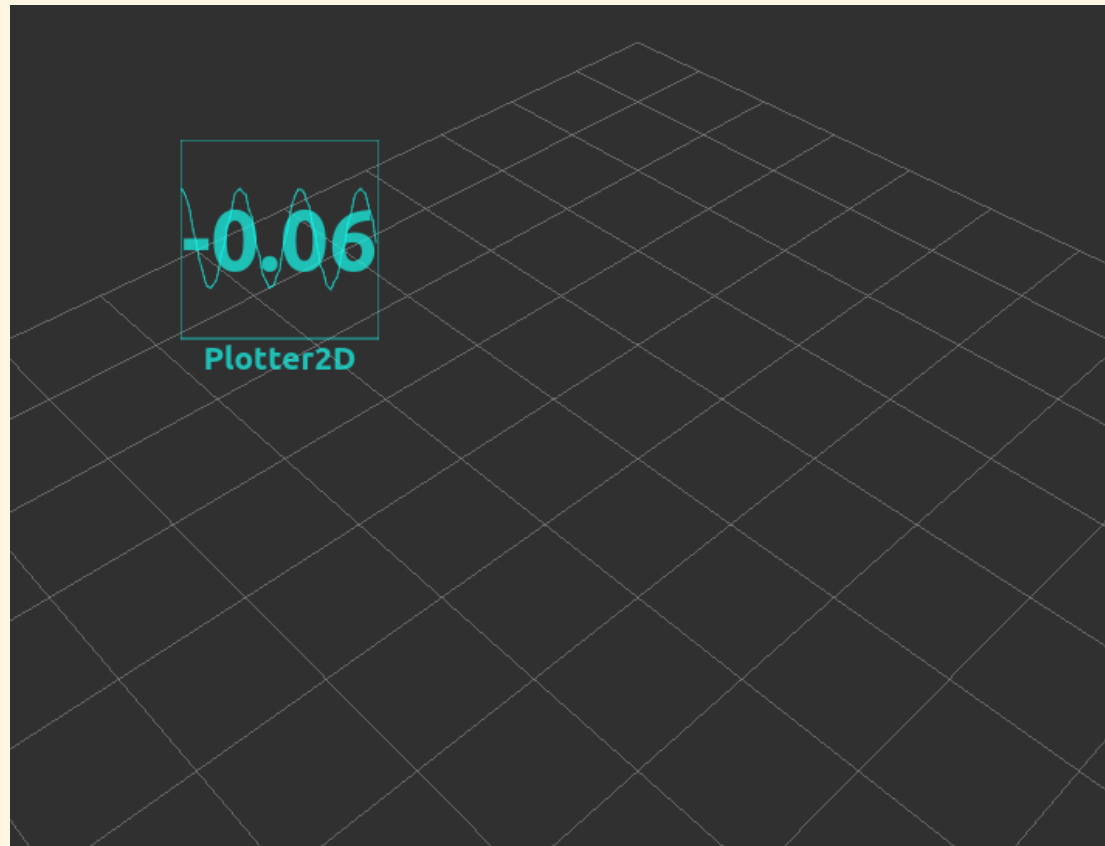
jsk_rviz_plugins OverlayText

- HUDでテキストを表示



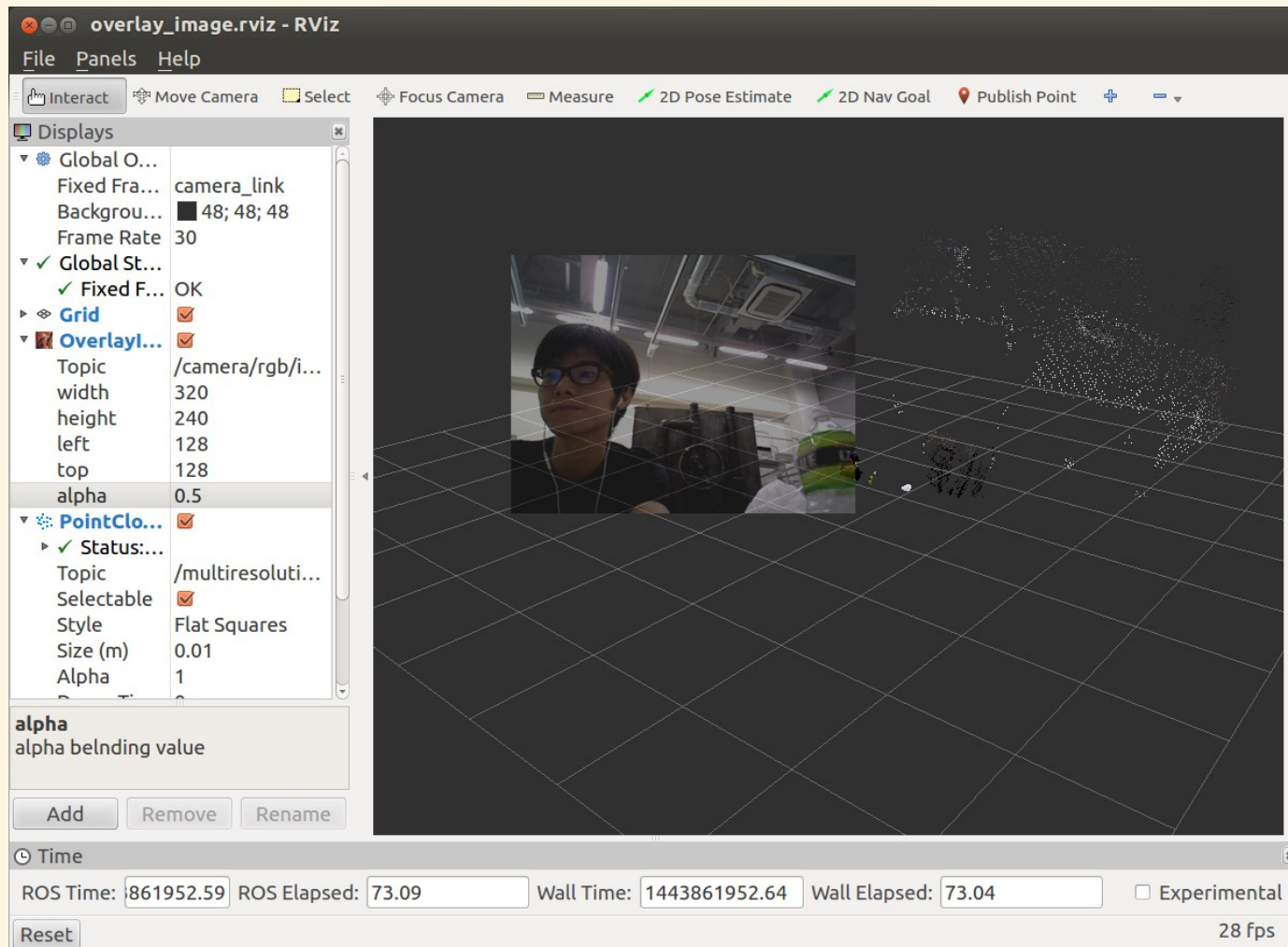
jsk_rviz_plugins Plotter2D

- HUDでグラフを表示



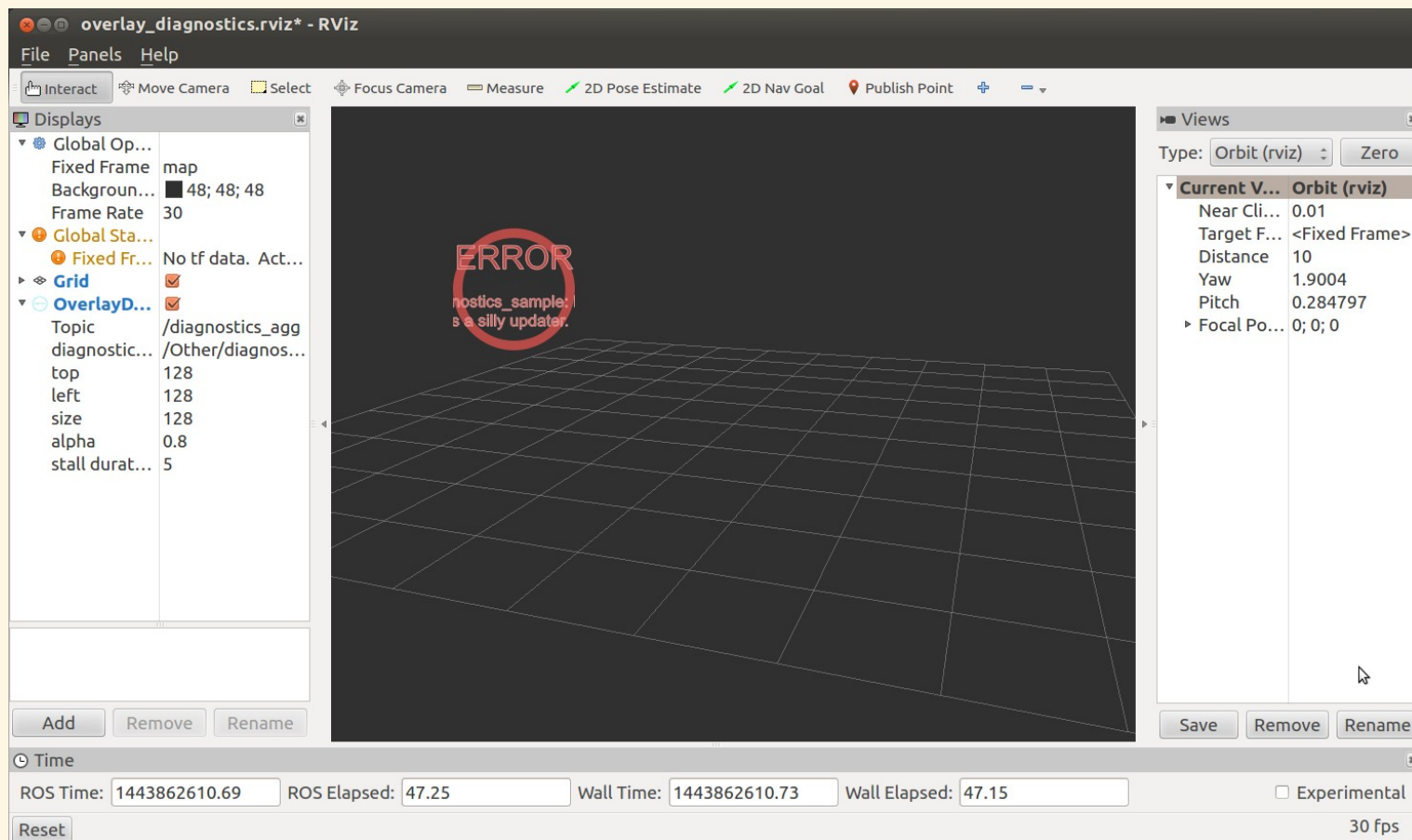
jsk_rviz_plugins OverlayImage

- HUDで画像を表示



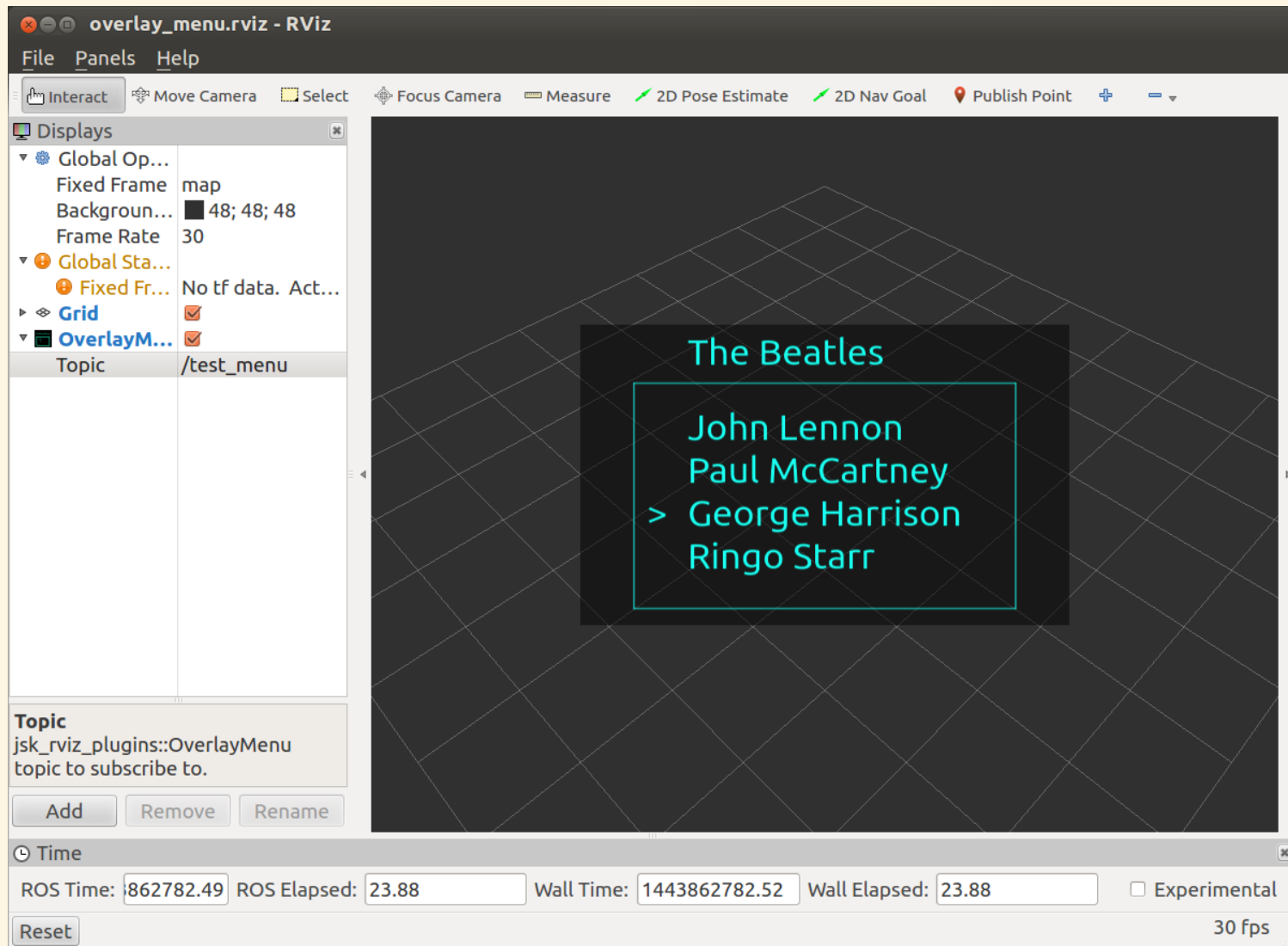
jsk_rviz_plugins OverlayDiagnostics

- HUDでDiagnosticsを表示



jsk_rviz_plugins OverlayMenu

- HUDでMenuを表示



おわりに: UI設計

- ROSでUIを作るときは、どこまでをrviz, rqtでやるかを考える
 - 基本的には、なるべくrvizでやると良い (panelをうまく使う)
 - Overlay系で2Dの可視化もrvizでできるならやったほうがいい(こともある)