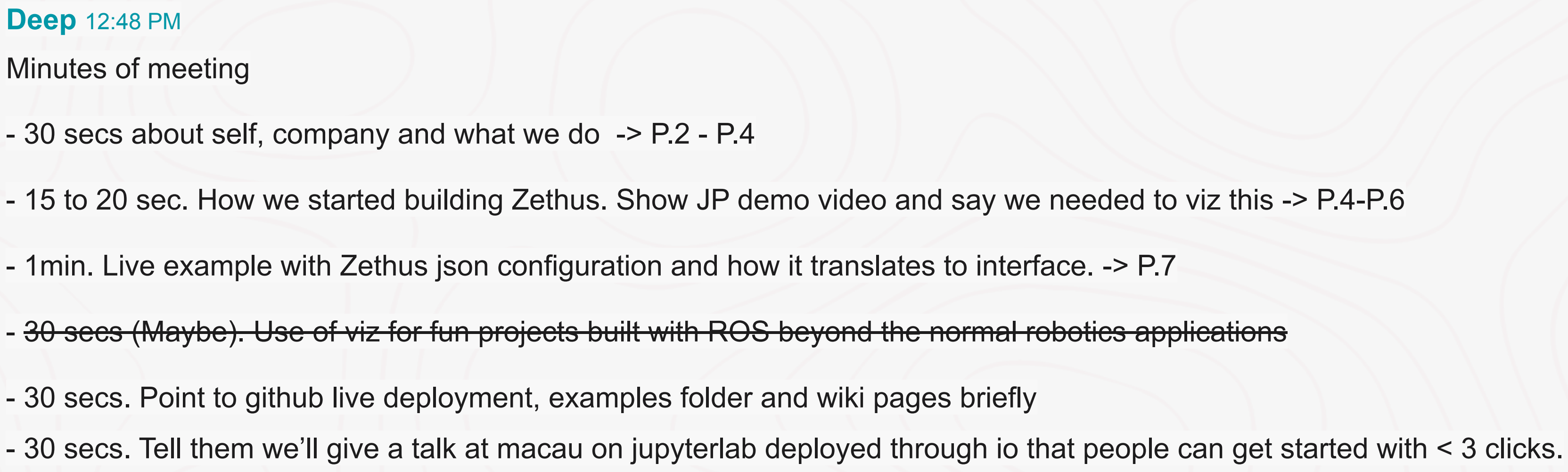
Agenda



Deep 12:48 PM

Minutes of meeting

- 30 secs about self, company and what we do -> P.2 - P.4

- 15 to 20 sec. How we started building Zethus. Show JP demo video and say we needed to viz this -> P.4-P.6

- 1min. Live example with Zethus json configuration and how it translates to interface. -> P.7

- ~~30 secs (Maybe). Use of viz for fun projects built with ROS beyond the normal robotics applications~~

- 30 secs. Point to github live deployment, examples folder and wiki pages briefly
- 30 secs. Tell them we’ll give a talk at macau on jupyterlab deployed through io that people can get started with < 3 clicks.

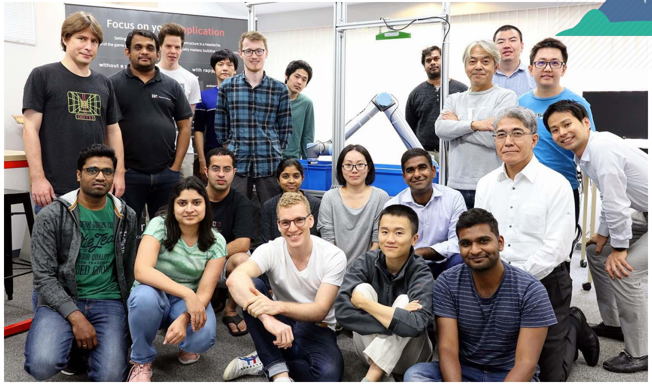
Rapyuta Robotics

ブラウザでのROS msg 可視化

strictly confidential copy

Rapyuta Robotics Overview

Team



Total funding: USD 24M

SONY　CYBERDYNE　三井住友信託銀行 SUMITOMO MITSUI TRUST BANK　SBI Investment

Awards

POST LOGITECH INNOVATION PROGRAM 最優秀賞 BEST AWARD JP 日本郵便

SoftBank Innovation Program

NEDO

J-Startup

Description

- ETH Zurich Spinoff　Spinoff ETH zürich
- 多国籍チーム：60数以上の国籍の約11名のメンバーが東京とインド（ベンガルール）で勤務
- 機械，電機，コンピュータ・サイエンス，ロボット，人工知能などの専門家が在籍

rapyuta.ioを介してロボットを連携

rapyuta.ioを介して連携

UNIVERSAL ROBOTS

rapyuta.io

Pickit
ROBOT VISION MADE EASY

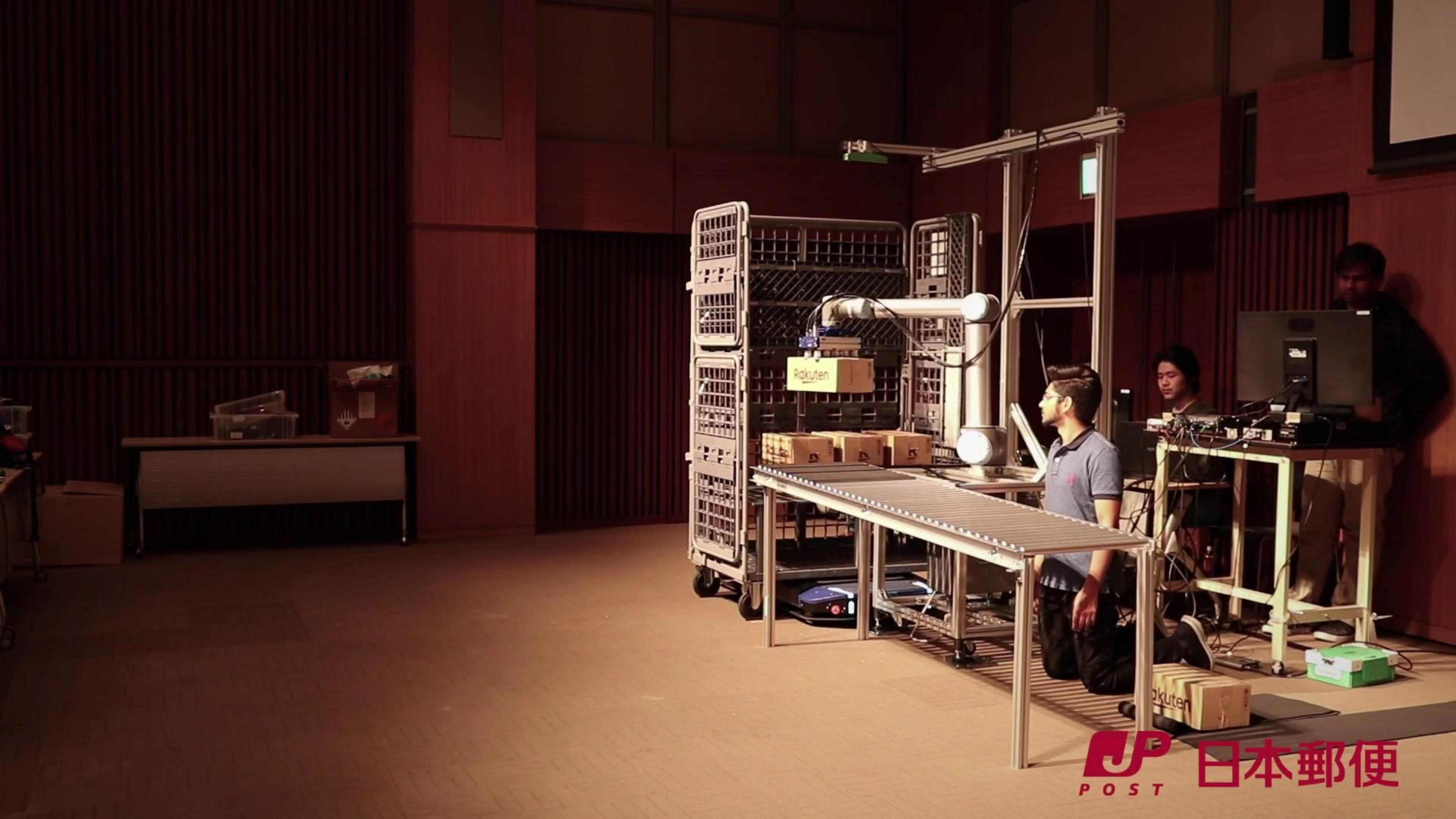
PICKNIK

SCHMALZ

KIANA
SYSTEMS

iFollow
collaborative robotics

JP POST 日本郵便


Rakuten
Rakuten
JP POST 日本郵便

必要になったWeb インターフェース

- ROSメッセージを可視化するWebインターフェース
- ロボット技術者でなくても使えるインターフェース
- 標準的なメッセージを高いパフォーマンスで表示
- 最小限の依存ライブラリで、Pluggableな可視化

Amphion: 可視化用Javascript ライブラリ

Zethus: Amphionを利用したROS msg可視化ブラウザインターフェス

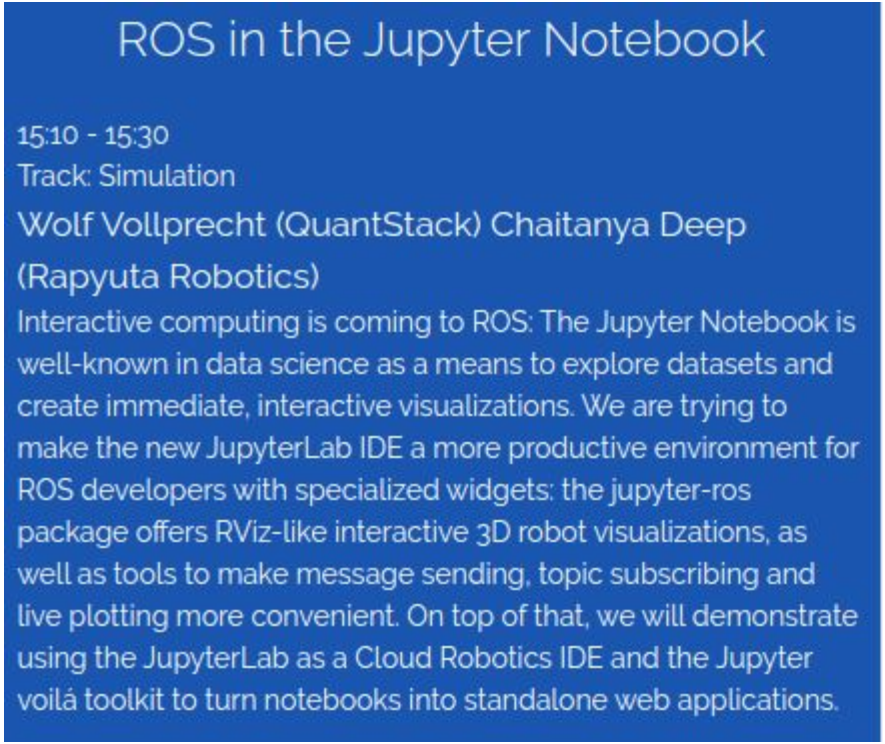
Kompose: Amphionを利用した可視化 + URDF編集 + moveit 制御まで含めたアプリ

Demo

Zethus and Amphion Opensource!!!

- github: https://github.com/rapyuta-robotics/zethus
- wiki: https://github.com/rapyuta-robotics/zethus/wiki
- examples: https://github.com/rapyuta-robotics/amphion/tree/devel/examples

Talk at ROSCon 2019



ROS in the Jupyter Notebook

15:10 - 15:30
Track: Simulation

Wolf Vollprecht (QuantStack) Chaitanya Deep (Rapyuta Robotics)

Interactive computing is coming to ROS: The Jupyter Notebook is well-known in data science as a means to explore datasets and create immediate, interactive visualizations. We are trying to make the new JupyterLab IDE a more productive environment for ROS developers with specialized widgets: the jupyter-ros package offers RViz-like interactive 3D robot visualizations, as well as tools to make message sending, topic subscribing and live plotting more convenient. On top of that, we will demonstrate using the JupyterLab as a Cloud Robotics IDE and the Jupyter voilá toolkit to turn notebooks into standalone web applications.

note from here

Why the existing solution (Ros3djs) didn't work

- It's **not a standalone/configurable application**, but rather a set of visualization objects intended to be used by developers.
- The visualization objects have **hard requirements**. Eg: Even loading a urdf requires a tf client. Creating a visualization automatically subscribes to message in the constructor.
- **Poor performance**. Eg: Updates to marker and marker array destroy and recreate the objects for each message. Odometry creates a new material for every message.
- **Less options and limited API**. One of the initial requirement was to load a simple urdf. There is an open-source project that is ros-independent, doesn't require a tf client and has functions to set joint values, load collision objects and the ability to have custom mesh loader
- **No active development**. There are simple issues on github that can be easily solved but open since 2017

Our solution

- A web interface (Zethus) that uses a **json configuration** for visualizations and options.
- Can be used **standalone**. Or included in other projects
- **React based**. Easy to integrate in other React apps. Can also be used in non-react applications with just the inclusion of React DOM
- Visualization components (Amphion) optimised for **performance**. Minimal updates, web assembly for intensive tasks. Other improvements in progress. Eg: Marker array.
- **Lot of options** for the visualizations. And a lot more in progress. Eg: Color schemes and min mag filters for map.

Where we are using it