Open Source Ag Robotics
Leveraging the power of ROS
History of the Robot Operating System

- ROS started as a personal project of Keenan Wyrobek and Eric Berger
- Willow Garage
- Open Source Robotic Foundation (OSRF)
Team MAXed Out

Learning Algorithms and Systems Laboratory (LASA) | EPFL
Survey

What about ROS convinced you to use it?

(multiple selections allowed)

- Communications infrastructure (middleware)
- License (open source)
- Community
- Integration with other projects (e.g., OpenCV)
- Developer tools
- Tutorials
- Drivers for your peripherals
- Support for multiple programming languages
- Documentation (non-tutorial)
- Your robot already runs ROS
- Your project / class required it

# responses
Sponsors
Background

PI for biological experiment on Space Shuttle Endeavor STS-77

Conducted Robotic Tractor Field Trials in Utah, Texas and California

Passionate about technology and the future

I ❤️ ROBOTS
ROS Agriculture

Vision
Autonomous farming.

Mission
Create an 'open source collaborative' community to reduce the barrier of entry for farmers in robotics.

An old proverb about collaboration - “If you want to go fast, go alone. If you want to go far, go together.”
Leveraging ROS
Tools
Tools
Tools
Capabilities

move_base Default Recovery Behaviors

Conservative Reset → Stuck → Clearing Rotation → Stuck → Aggressive Reset → Stuck → Clearing Rotation → Stuck → Aborted

Navigating

clear → clear → clear

No obstacle

With obstacle
Capabilities

Movelt!
Capabilities
# Capabilities

## ROS Index

---

### Packages

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>grid_map_core</td>
<td>Universal grid map library to manage two-dimensional grid maps with multiple data layers.</td>
</tr>
<tr>
<td>grid_map</td>
<td>Meta-package for the universal grid map library.</td>
</tr>
<tr>
<td>generic_throttle</td>
<td>This package provides a throttle for ROS topics.</td>
</tr>
<tr>
<td>dockeros</td>
<td>Simply running ros nodes in docker containers on remote robots.</td>
</tr>
<tr>
<td>dhw_mkz_twist_controller</td>
<td>Twist (speed and angular rate) controller for brake/throttle/steering</td>
</tr>
<tr>
<td>dhw_mkz_msgs</td>
<td>Drive-by-wire messages for the Lincoln MKZ.</td>
</tr>
<tr>
<td>dhw_mkz_joystick_demo</td>
<td>Demonstration of drive-by-wire with joystick</td>
</tr>
<tr>
<td>dhw_mkz_description</td>
<td>URDF and meshes describing the Lincoln MKZ.</td>
</tr>
<tr>
<td>dhw_mkz_can</td>
<td>Drive-by-wire interface to the Dataspd Inc. Lincoln MKZ DBW kit</td>
</tr>
<tr>
<td>dhw_mkz</td>
<td>Drive-by-wire interface to the Dataspd Inc. Lincoln MKZ DBW kit</td>
</tr>
<tr>
<td>ccr_multimaster_tools</td>
<td>Multi-master tools for configuration and message relaying</td>
</tr>
<tr>
<td>cob_teleop</td>
<td>Teleop node</td>
</tr>
<tr>
<td>cob_script_server</td>
<td>The cob_script_server package provides a simple interface to operate Care-O-bot. It can be used via...</td>
</tr>
<tr>
<td>cob_monitoring</td>
<td>cob_monitoring</td>
</tr>
<tr>
<td>cob_interactive_teleop</td>
<td>COB teleop interactive marker for RViz provided by dgsm-robotics@FIT group.</td>
</tr>
<tr>
<td>cob_helper_tools</td>
<td>Helper scripts for Care-O-bot</td>
</tr>
<tr>
<td>cob_dashboard</td>
<td>cob_dashboard is a modified version of [[pr2_dashboard]].</td>
</tr>
<tr>
<td>cob_command_tools</td>
<td>The cob_command_tools stack provides tools, for operating Care-O-bot.</td>
</tr>
<tr>
<td>cob_command_gui</td>
<td>This package provides a simple GUI for operating Care-O-bot.</td>
</tr>
<tr>
<td>clock Relay</td>
<td>Specialization of message_relay for clock</td>
</tr>
</tbody>
</table>
**Ecosystem - Open Community**

**ROS Tutorials**

*Non-Beginners:* If you're already familiar enough with ROS fuerte or earlier versions and only want to explore the new build system introduced in groovy and used in hydro and later, called catkin, you can go through more in-depth [catkin tutorial here](#). However, going over all basic Beginner Level tutorials is still recommended for all users to get exposed to new features.

*If you are new to Linux:* You may find it helpful to first do a quick tutorial on common command line tools for linux. A good one is [here](#).

**Contents**

1. Core ROS Tutorials
   1.1 Beginner Level
      1. Installing and Configuring Your ROS Environment
         This tutorial walks you through installing ROS and setting up the ROS environment on your computer.
      2. Navigating the ROS Filesystem
         This tutorial introduces ROS filesystem concepts, and covers using the roscd, rosls, and rospack commandline tools.
      3. Creating a ROS Package
         This tutorial covers using roscat or catkin to create a new package, and rospack to list package dependencies.
      4. Building a ROS Package
         This tutorial covers the toolchain to build a package.
What's New?

- Communication backbone (DDS)
- Security
- Multi Robot Swarms
- Real-Time Support
- more...
Certified ROS

Apex.AI™

Apex.OS™
Safe and certified software framework and SDK for autonomous mobility systems.

Apex.OS abstracts complexity of underlying hardware, middleware, kernel, interfaces, and drivers into simple to use, robust, reliable, safe, secure APIs.
Matt Droter
matt@rosagriculture.org
Text ‘ros’ to 345345