

Introduction to ROS



Wyatt Newman

Case Western Reserve University

EECS Dept

NASA/Goddard

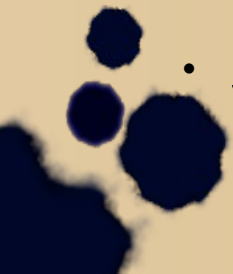
July 9-10, 2015



Why ROS (Robot Operating System)?



- Stagnation in robotics. Need for:
 - Code re-use
 - Standard interfaces that support problem decomposition
 - Architecture that exploits distributed development
 - Architecture that exploits distributed computing (network or cloud)
 - Means to share code (repositories)
 - Scalable



Some prior attempts



- RCCL (Hayward and Paul, Purdue, '84)
- National Instrument (LabView robotics; FIRST)
- Microsoft Robotics Developer Studio ('06).



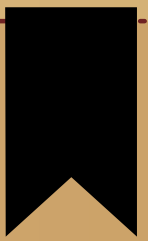
What is ROS and why is it different?



- STAIR 1: STanford Artificial Intelligence Robot 1.
 - “Morgan Quigley created the Switchyard framework to provide a robot framework for their mobile manipulation platform, and it was the lessons learned from building software to address the challenges of mobile manipulation robots that gave birth to ROS.”
- ROS history:
 - Stanford AI lab robotics (STAIR) project, through 2007
 - Google support:
 - Willow Garage, 2008-2013 (introduced ROS 1.0)
 - Open Source Robotics Foundation (OSRF, 2013-present)
 - ROS releases to date through “Jade” (9th version)
 - Repository support, Q&A, ROSCON
 - Open source, freeware



What is ROS?



- Not really an “operating system”
- Per Brian Gerkey:

1. plumbing: ROS provides publish-subscribe messaging infrastructure designed to support the quick and easy construction of distributed computing systems.
2. tools: ROS provides an extensive set of tools for configuring, starting, introspecting, debugging, visualizing, logging, testing, and stopping distributed computing systems.
3. capabilities: ROS provides a broad collection of libraries that implement useful robot functionality, with a focus on mobility, manipulation, and perception.
4. ecosystem: ROS is supported and improved by a large community, with a strong focus on integration and documentation. ros.org is a one-stop-shop for finding and learning about the thousands of ROS packages that are available from developers around the world.



What is ROS (cont.)?



- From the ROS wiki:
 - The primary goal of ROS is to support code reuse in robotics research and development. ROS is a distributed framework of processes (aka “nodes”) that enables executables to be individually designed and loosely coupled at runtime. These processes can be grouped into Packages, which can be easily shared and distributed. ROS also supports a federated system of code Repositories that enable collaboration to be distributed as well. This design, from the filesystem level to the community level, enables independent decisions about development and implementation, but all can be brought together with ROS infrastructure tools.



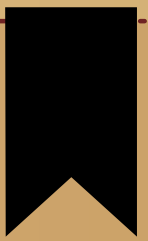
What is ROS (cont)?



- From the text ROS by Example:
 - The primary goal of ROS (pronounced "Ross") is to provide a unified and open source programming framework for controlling robots in a variety of real world and simulated environments.
 - The core entity in ROS is called a node. A node is generally a small program written in Python or C++ that executes some relatively simple task or process. Nodes can be started and stopped independently of one another and they communicate by passing messages. A node can publish messages on certain topics or provide services to other nodes.



ROS and the DARPA Robotics Challenge



- (unofficially) every team that qualified for the DARPA Robotics Challenge Finals used ROS
 - (JPL used ROS in part)
 - Accelerated schedule:
 - Started building robots Oct '12
 - Trials Dec '13
 - Finals Jun '15
- Example DRC applications with ROS (Newman/TeamHKU):
 - Valve turning
 - Wall cutting





Autonomous Valve Turning with an Atlas Humanoid Robot

The University of Hong Kong

Wynne Newman, Senior Member IEEE, Zheng Han Cheng, Cong Shi,
Robert T. W. Hung, Kit-Hang Lam, Wu Li, Tony W. S. Ng,
Christopher L. Sweetman, Kenneth K. S. Tseng, Waike Wong

Autonomous Wall Cutting with an Atlas Humanoid Robot



Faculty of Engineering
THE UNIVERSITY OF HONG KONG

Advanced Robotics Initiative

Wyatt Newman, Senior Member IEEE, Zheng-Hao Chong, Robert T.W. Hung, Kit-Hang Lee, Tony W.L. Ng, Christopher E. Swetenham, Kenneth K.S. Tjoeng, Weijia Wang

Contact: robotics@hku.hk

A ROS demonstration:



- Baxter simulation with supervisory control

