Peripherals

Beyond Ros Core Daniel Stonier Yujin Robot

Contents

- Technologies
- Interactions
- Multi-Robot-Device
- Korean Group ETRI

Connecting Technologies

Reusable tools and building blocks for ros with qt.

RQT

Java/Android

Bridging to a java-based ecosystem

OR

Hand held interactions with androids

Web Tools

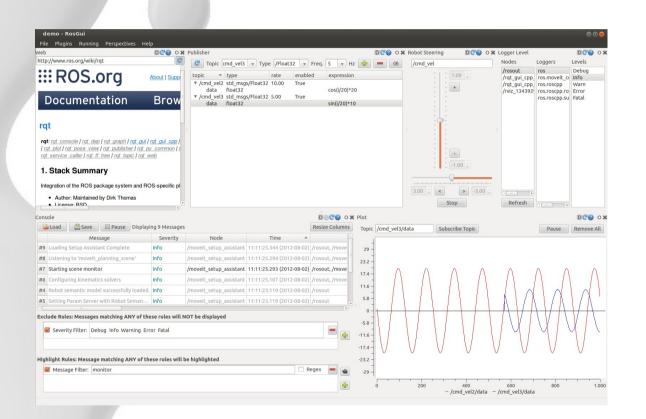
Device agnostic ui's







RQT



The Official Graphical Framework for Ros Tools

> C++ or Python Your Choice!

Lots of Reusable Modules and Widgets

Full Access to Ros Core API and Tooling

Completely integrated catkin workflow

Low Learning Curve and Lots of Support Potential to be cross-platform (linux-mac-win-tablet) but not enough critical mass



http://wiki.ros.org/rqt

Java/Android

Reasonably Complete Comms and Core Libraries

New to Hydro

Java Debs Individual Message Jars Maven Repo On Github Better Gradle Integration

The Android Studio Revolution!

What Does This Mean???

Build Side By Side with Ros (Catkin Make) OR Build Standalone Without Ros (Maven Repos)



Manage a Collection of Android Libraries (.AARs)

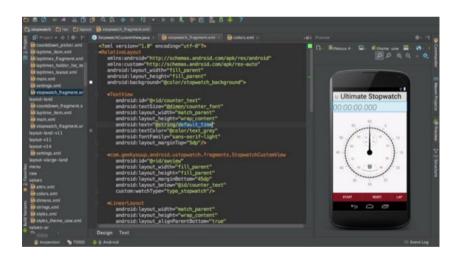
Build an Android App without a Complicated Ros Install



Continuous Integration for Android Development

Coming to Indigo

Native Message Generation



Still Very Ros-Like

Missing Some Transports (Actions) Not Currently Being Developed

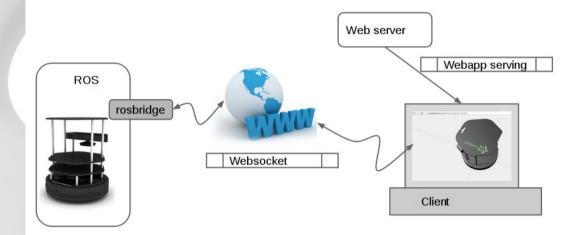


http://wiki.ros.org/rosjava

http://wiki.ros.org/android

Web Tools

How It Works



Rapidly Growing Collection

Active Development

Zero Installation (User Side)

Very Cross Platform Embeddable In Other Web Applications Requires a Bridge Can't Easily Access the Device/PC



http://www.robotwebtools.org/

mailto:robot-web-tools@googlegroups.com

Comparison

RQT	Android	Web Apps
For Ros Devs	For Java Devs	For SW/Web Devs
Quick Prototyping & Low Maintenance	lf you need the device NFC Speed	No Installation for Users
Can do very complex lots of library support	Can do complex algorithms scientific support	Embeddable in Web Application Suites
Dev Tools Factory SW Prototype UI Unstable UI	Android NFC Apps Web App Launchers	Customer Frontends, Embedded Web Apps, Remote Monitoring



@Yujin

Rocon

Capabilities

Laying down structure at the bootstrap level of a ros robot.

Robot App Management

A public task-like interface to your ros robot.



Interactions

Start interacting with your ros robot without needing ssh connections or remembering long uri's.



Robot Capabilities

Q) "I need Navigation..."

What software stack? What is the interface specification?

How do I launch it with my software? What if I want to depend on a modified software stack?

Building Capabilities

Standard Capabilities (Interfaces)

> Capability Providers (Implementations)

Runtime - The Capability Server

Start/Stop Capabilities

Remap the ROS api

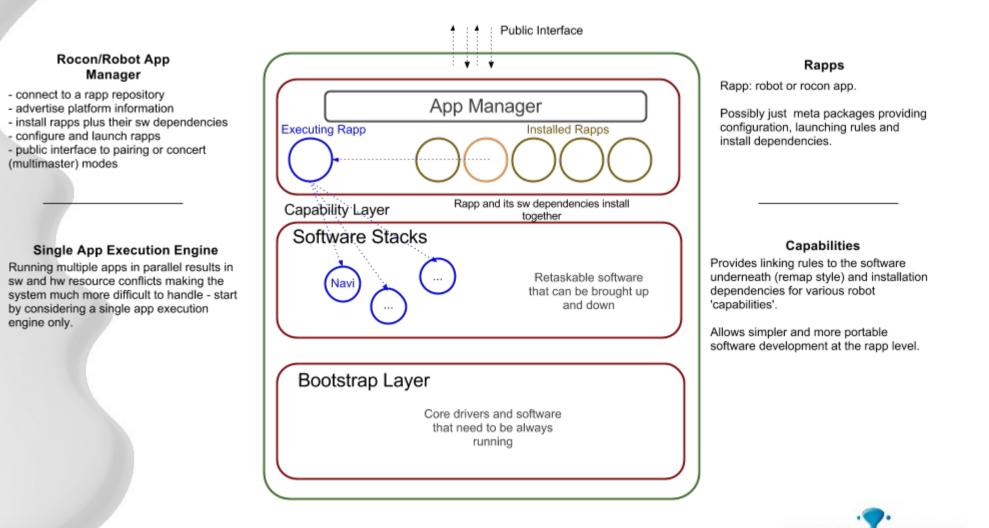
Semantic Capabilities (head image, side image)



Application Management

A Task-Like Interface for Your Robot

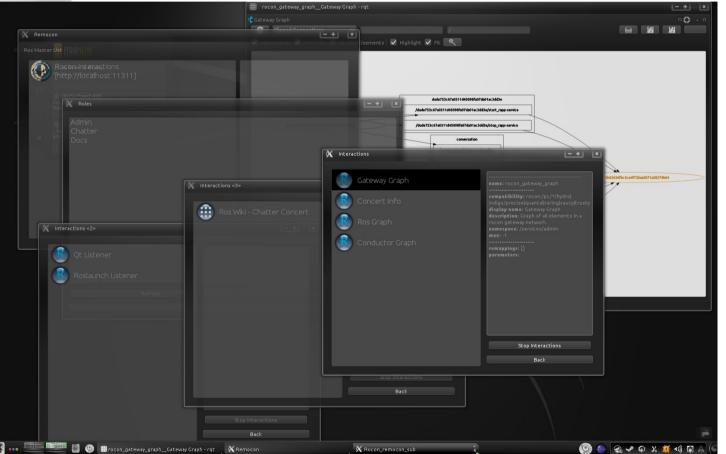
Tear Down and Build Up Entire Launch Environments



http://wiki.ros.org/rocon_app_platform

Interactions

Human-Ros Master Software Interactions



Remocons - Interaction Launchpads

Android & Qt Remocons

Interaction Types

Qt Tools Android Apps Web Apps Documentation Links RosRunnables RosLaunchers

Parameterise and/or Remap Launches

Pair With Robot App Managers



http://wiki.ros.org/rocon_interactions

Concert

Gateways

Distributed Multimaster - each ROS robot as a gateway on a hub.



Orchestration

Is there a right way to orchestrate/choreograph multiple robots?



Gazebo Concert

And you thought debugging one ROS robot was hard? Enter gazebo!!!

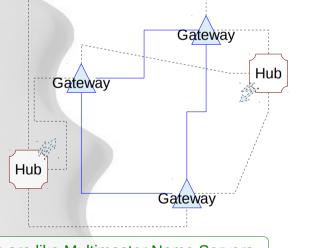


Gateways

Multimaster Communications

- True multimaster
- Hides Ros Masters
- Public Interfaces
- Firewalls
- Auto-Discovery

(Zeroconf) Each Gateway is a Door to a Ros Master



Hubs are like Multimaster Name Servers

Now with Robust Wireless Handling

Flips & Advertisements/Pulls

- Control where topics are sent with **flips**
- Share freely with adverts &
 pulls

Gateway

Gateway

Master

Flipping



Master

rules placed on watchlists.

Gateways act on them as they locally appear and disappear.

Master GatewAdverts /chatter Adv & Pull

Build Your Own Multimaster System on Top of the Gateways

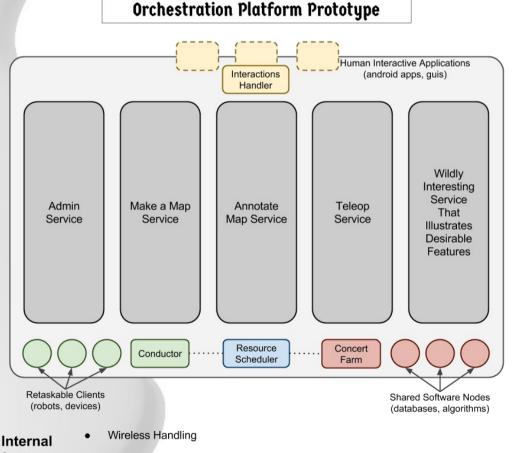


http://wiki.ros.org/rocon_gateway

Orchestration

Orchestrating Services with Multiple Robots-Devices-Humans

There is no Right Way!!! Yet... So Much Infra is Needed...



features • Multiple Service Paradigms (ROS, Static Link Graphs, Orc, BPEL)

We Decided:

Parallel Services (Like a Web Server)

Each service is an independent orchestration block

Give services the freedom to do their own orchestration

We Just build the Infra Around the Services Communications, Robot Task Scheduling, Software Launching, Human Interactions,



http://wiki.ros.org/rocon/indigo/Guide

Gazebo Concert

/mnt/zaphod/ros/segbots : python	concert:11311	<u> </u>			<u> </u>
file Edit View Bookmarks Setting norri@snorriheim:~\$ cdros norri@snorriheim:/mnt/zaphod/ros\$	File Edit View Bookmarks Settings Help rocon:/:['guimul'] c/Prund Tree		File Edit View Bookmarks Settin [DEBUG] [1401907791.689138332, 76 [DEBUG] [1401907791.689291983, 76	ings Help 6.371000000]: Got a valid command from th 6.371000000]: Full control cycle time: 0.	he local planner: 0.774, 0.000, 0.007 🖌
norri@snorriheim:/mnt/zaphod/ros/: ndroid-studio build config.cmak	egbo rocoh:/ : ['guimul'] [INFO] [WallTime: 14019077261749268]nScheduler : compatibility tree i:	is valid, attempting to allocate [4ca49a18-02	207 [DEBUG] [1401907791.728557589, 76		
aunching /mnt/zaphod/ros/segbots/	-4309-85cc-b5f7el15e7a1]:ot pazebo concert gazebo concert -screen [INFO] [WallTime: 1401907727.745754] Conductor : concert client trans [INFO] [WallTime: 1401907720.884197] Scheduler : allocated [guinul] [INFO] [WallTime: 1401907720.884197] Scheduler : anaveconcert client [Compatibility: Treephot gazebo concert/solutions/concert.aunch on por remons/sol 1400bit 10 Concept) in concert of an other scheduler.	sition [joining->əvailable][kobuki] [kobuki] ort 11311	58 and stamp: 72 86	5.414000000]: Publishing feedback for goa 5.414000000]: In controlling state. 6.414000000]: Got a valid command from th 6.414000000]: Full control cycle time: 0.	al with id: /kobuki/segbot_stroll-5-72.8 he local planner: 0.774, 0.000, 0.008
DBusConnection: session D-Bus con (ConfigIni: In file /home/snorri/	<pre>lecrocon:/estidkobukke/UCoreApplication. Application may misbehave. k/Pruning.icons/Iceglass 4.0/index.theme, line 30: " Invalid entry (m: rocore.d. ('kobuki')</pre>			6.414000000]: Full control cycle time: 0. 6.477000000]: Publishing feedback for goa	
🞐 Gazebo : default	Pruning Resolvable Branches rocon:/ : ['kobuki']		stamp: 72,86		al with id: /kobuki/segbot_stroll-5-72.8
ile <u>E</u> dit <u>V</u> iew <u>W</u> indow <u>H</u> elp	<pre>Pruned Tree rocon:/: ['kobuki'] [INFO] (WallTime; 1401907730.905599] Scheduler : compatibility tree i</pre>	is valid, attempting to allocate [eddf24cc-67	58 and stamp: 72.86 [DEBUG] [1401907791.794343400, 76 [DEBUG] [1401907791.795116472, 76	6.477000000]: In controlling state. 6.4770000001: Got a valid command from th	he local planner: 0.774. 0.000. 0.010
Vorld Insert	-41c6-bb7c-3a50d5b071b6] [INF0] [WallTime: 1401907735.661704] Scheduler : allocated [kobuki] []		C C C C C C C C C C C C C C C C C C C	6.477000000]: Full control cycle time: 0.	.000975262
│ Physics > Models > Lights	concert:11311		kobuki:114	411	
Lights			guimul:11412		- +
			[DEBUG] [1401907791.685471947, 76	ings Help 6.371000000]: In controlling state. 6.371000000]: Got a valid command from th 6.371000000]: Full control cycle time: 0.	he local planner: 0.774, 0.000, -0.017
			nd stamp: 69.01		
operty Value			[DEBUG] [1401907791.727798414, 76 [DEBUG] [1401907791.727972389, 76 [DEBUG] [1401907791.728094332, 76	6.414000000]: In controlling state. 6.414000000]: Got a γalid command from th 6.414000000]: Full control cycle time: 0.	he local planner: 0.774, 0.000, -0.018 .000492940
			Contraction of the second seco		
			nd stamp: 69.01		
			[DEBUG] [1401907791.792375484, 76 [DEBUG] [1401907791.792546036, 76 [DEBUG] [1401907791.792598277, 76	6.477000000]: In controlling state. 6.477000000]: Got a valid command from th 6.477000000]: Full control cycle time: 0.	he local planner: 0.774, 0.000, -0.019 .000375466
			auimul:11	1412	
		\times			
	Steps: 1 Real Time Factor: 0.88	Sim Time: 00 00:01;16.359 Real Time: 00 00:01:32.	2.665 Iterations: 76359 Reset		
1 🐼 💷 💶 🍡	szebo : default	Sim Time: 00 00:01:16.359 Real Time: 00 00:01:32.			ک 🖛 🖷 شنه که 💥 س) 🗊 ۵ AM 3:4

Ros in Korea

Electronics and Telecommunications Research Institute ETRI Co-Pilot System with Ros YongBon Koo Senior Researcher Autonomous Driving System Research Section





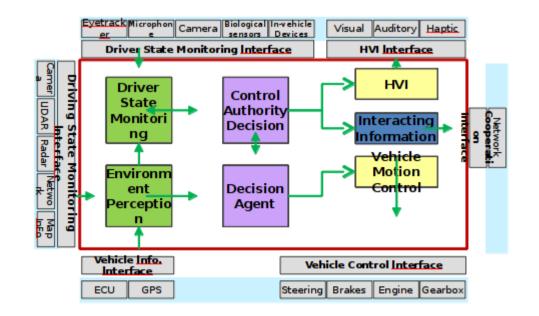
ETRI Co-Pilot System

Co-Pilot System is

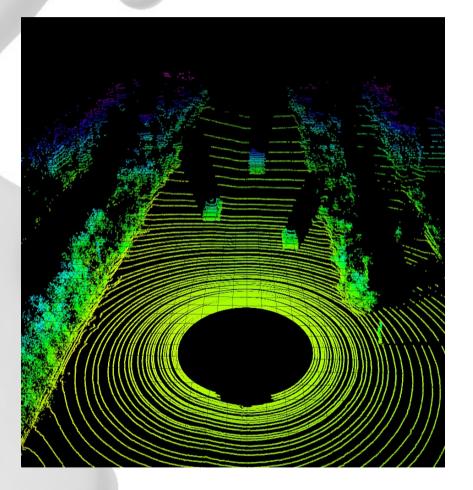
- Basically, a fully autonomous vehicle system
- Like a google driverless car
- Human and autonomous vehicle cooperation

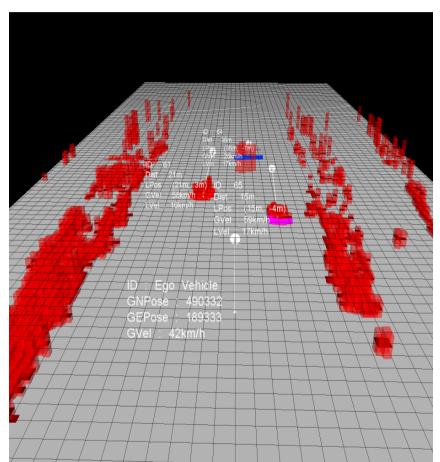
Ros for Co-Pilot System

- Is a IPC framework between the b
- Offers some basic components



Environment Resception



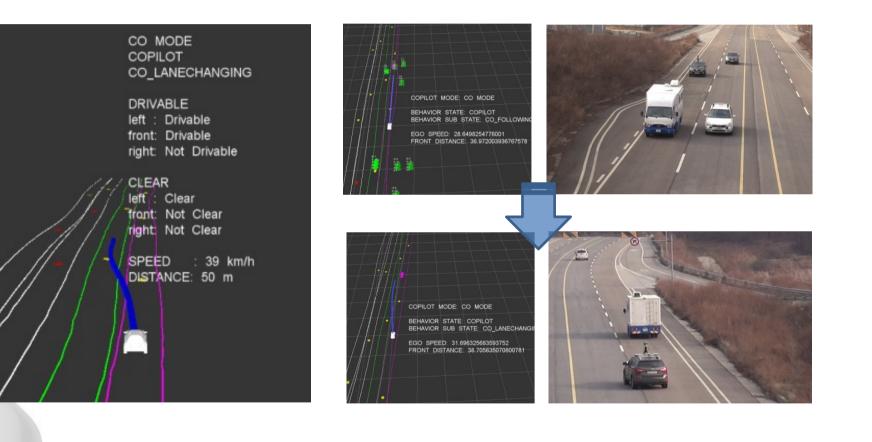




Decision Agent

Based on Existing Univ. of Texas-Austin Code

Map Lanes, Observers





ETRI – Why We Chose ROS

Ros is Great For

- Who wants to make a working prototype rapidly
- You can use many components for
 - Communication, Sensor, Path Planning
- Fantastic for
 - Visualization, I love it!
 - Logging and Replay I can't live without it!

But it Lacks

- Support for other platforms
 - We need ROS for LabView, Matlab, OSX, iOS.
- Documentation
 - Make it together

