ROS Android-NDK

What? Why? How?

Gary Servín

Ekumen

#ROSCon15
October 3, 2015
Our team
Our clients

- Open Source Robotics Foundation
- Willow Garage
- Savioke
- Yujin Robot
JavaTM Native Interface (JNI)
a standard programming interface for executing native code inside Java applications.

Android NDK
a tool-set that allows you to implement parts of your app using native-code languages such as C and C++.

rosjava (http://wiki.ros.org/rosjava?distro=indigo)
an implementation of ROS in pure Java with Android support.
It’s a set of scripts that make easier to cross-compile ROS packages/libraries using the Android NDK.

This allows to run ROS packages/libraries on Android as:

• a native activity (C++),
• a normal app using the JNI and rosjava
• a native executable
What’s been done?

• 181 ROS packages from the Desktop variant (-viz -python)
• Tutorials (http://wiki.ros.org/android_ndk)
  • Native Activity
  • JNI wrappers with rosjava
• robot_localization demo
Why is it interesting?

Android devices are becoming more popular and ubiquitous

Powering screens of all sizes

Android is the customizable, easy to use operating system that powers more than a billion devices across the globe — from phones and tablets to watches, TV, cars and more to come.
Why is it interesting?

Android devices are becoming more powerful
Why is it interesting?
Android devices are becoming cheaper
Why is it interesting?
Why not just use rosjava?

Source: https://xkcd.com/927/

**How standards proliferate:**
(See: A/C chargers, character encodings, instant messaging, etc.)

**Situation:**
There are 14 competing standards.

14?! RIDICULOUS! We need to develop one universal standard that covers everyone's use cases. YEAH!

**Soon:**
Situation: There are 15 competing standards.
Why not just use rosjava?

• No need to re-implement functionalities in Java
• Can still be used together with rosjava
• run ROS nodes native directly (su required) or via JNI
Future work

- better way to get and update ROS packages (Done)
- output order of the libraries needs to be adjusted manually
- rosjava wrappers
- rospack (python and filesystem structure)
- pluginlib (hardcoded plugins)
- actionlib
- test packages!
- ROS 2.0 support
How to contribute?

- http://wiki.ros.org/android_ndk/
- https://github.com/creativa77/roscpp_android
- gary@ekumenlabs.com | @garyservin
- http://www.ekumenlabs.com/