BLG456E
Robotics
ROS Intro

Lecture Contents

ROS overview.
Other middlewares.
ROS distributions.
Installing ROS & Turtlebot simulation.
Directories & variables.
Tutorials.
ROS build system.
ROS nodes.

Lecturer: Damien Jade Duff
Email: djduff@itu.edu.tr
Office: EEBF 2316
Schedule: http://djduff.net/my-schedule
Coordination: http://ninova.itu.edu.tr/Ders/4709

ROS Overview

- ROS = "*Robot Operating System*"
  - Not an operating system!
- Contains:
  - Middleware (loosely coupled publish-subscribe).
  - Build and packaging system.
  - Core and peripheral packages.
    - E.g. reference frame management, map management, simulators, device drivers, controllers, navigators, recognizers, etc.

Robot Middleware

- Main contenders:
  - ROS (Robot Operating System)
  - YARP (Yet Another Robot Program)
  - Player/Stage
  - Dozens of others (including CAST).

Pic: Willow Garage

ROS Distributions

- Distributions rolled out approx half-yearly (future: yearly):
  - Indigo Igloo July 22nd, 2014.
  - Hydro Medusa September 4th, 2013.
  - Groovy Galapagos December 31, 2012 .
  - Fuerte Turtle April 23, 2012.
  - Electric Emys August 30, 2011.
  - Diamondback March 2, 2011.
  - C Turtle August 2, 2010.
  - Box Turtle March 2, 2010.
- Built on Ubuntu.
  - Other platforms available, not as reliable.
  - **Not all ROS distributions work on all Ubuntus.**
    - **ROS Indigo Igloo → Ubuntu 14.04**
  - http://wiki.ros.org/fuerte/Installation/Ubuntu

BLG456E
Robotics
ROS Intro

Lecture Contents

ROS overview.
Other middlewares.
ROS distributions.
Installing ROS & Turtlebot simulation.
Directories & variables.
Tutorials.
ROS build system.
ROS nodes.

Lecturer: Damien Jade Duff
Email: djduff@itu.edu.tr
Office: EEBF 2316
Schedule: http://djduff.net/my-schedule
Coordination: http://ninova.itu.edu.tr/Ders/4709

Installing ROS & Turtlebot

- Step 1: Install Ubuntu.
  - Option 1: Hard disk install.
  - Option 2: Virtual machine install (slow).
  - Option 3: External hard disk install.
- Step 2: Install ROS.
  - Run shell script **install_ros_indigo.sh** from https://bitbucket.org/damienjadeduff/456_indigo_turtlebot/src
- Step 3: Install the Turtlebot simulation.
  - Use shell script **install_turtlebot_gazebo.sh** from https://bitbucket.org/damienjadeduff/456_indigo_turtlebot/src

Detailed instructions can be found on Ninova under Course Files / Ders Kaynaklar

Directories & variables

- ROS Indigo installed in `/opt/ros/indigo`
- To make use of it:
  - Run from a terminal:

    ```
    source /opt/ros/indigo/setup.bash
    ```

  - Add that to `~/.bashrc` to make it permanent (**install_turtlebot_gazebo.sh** does this for you).

How to use ROS

- Follow the tutorials to learn the basics.
  - Make sure to select the version of ROS that you are using (e.g. Indigo).
- Suggested order of tutorials for intro to ROS:
  - Installing & Configurating your ROS Environment
  - Navigating the ROS Filesystem
  - Creating a ROS Package
  - Building a ROS Package
  - Understanding ROS Nodes
  - Understanding ROS Topics
  - Writing a Simple Publisher & Subscriber (C++)
  - Examining the Simple Publisher & Subscriber
  - Using rqt_console & roslaunch

BLG456E
Robotics
ROS Intro

Lecture Contents

ROS overview.
Other middlewares.
ROS distributions.
Installing ROS & Turtlebot simulation.
Directories & variables.
Tutorials.
ROS build system.
ROS nodes.

Lecturer: Damien Jade Duff
Email: djduff@itu.edu.tr
Office: EEBF 2316
Schedule: http://djduff.net/my-schedule
Coordination: http://ninova.itu.edu.tr/Ders/4709

ROS concepts:
ROS build-system built

- Functionality comes in “packages”.
- Existing functionality installed by package.
- **Your** programs will be built as packages.
- Toolchains for building:
  - rosbuild – older, deprecated.
  - catkin – newer, cmake-based

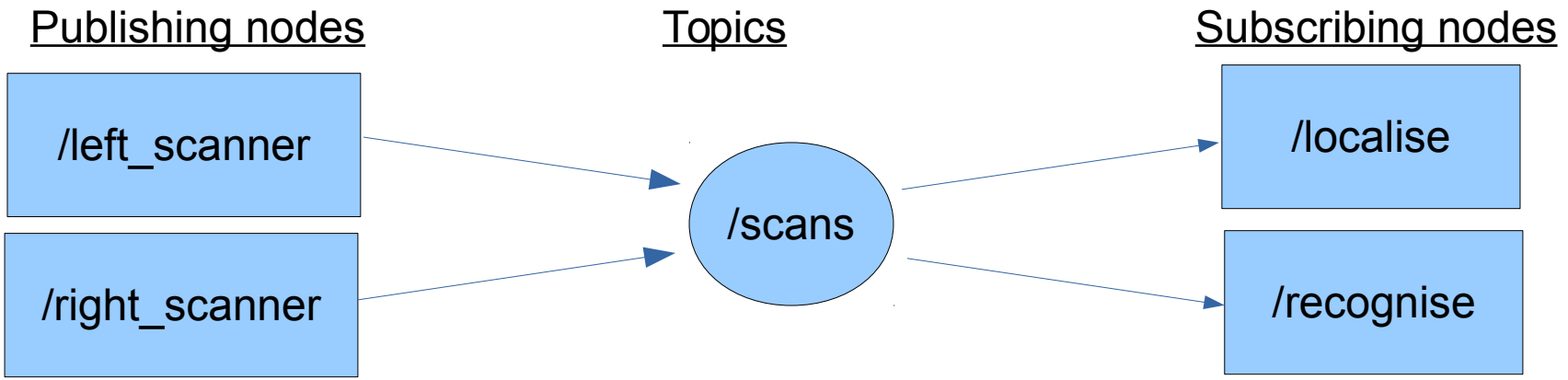
ROS concepts:

A ROS program is a set of communicating nodes.

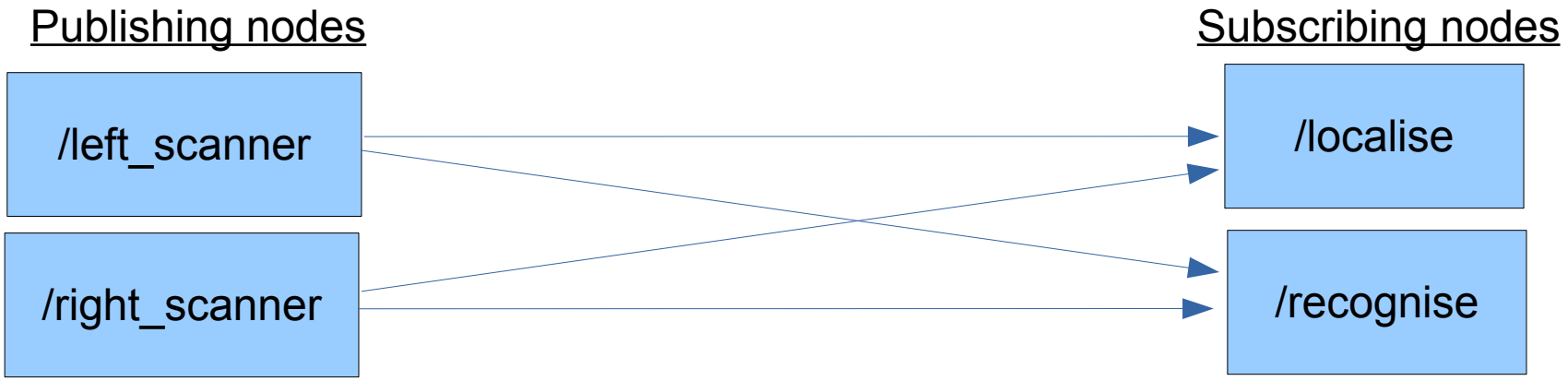
- ROS uses a publish-subscribe model.
- Programs construct nodes.
- Publishing nodes send messages to a topic.
- Subscribing nodes take messages from a topic.

ROS Node graph

Conceptual node graph



Graph of message routes



ROS concepts:

A ROS program is a set of communicating nodes.

When nodes connect:

- Publishing nodes contact the ROS master program over TCP/IP.
- ROS master gives the address of all nodes subscribed to that topic.
- Publishing nodes sends all messages directly to subscribing nodes over TCP/IP.