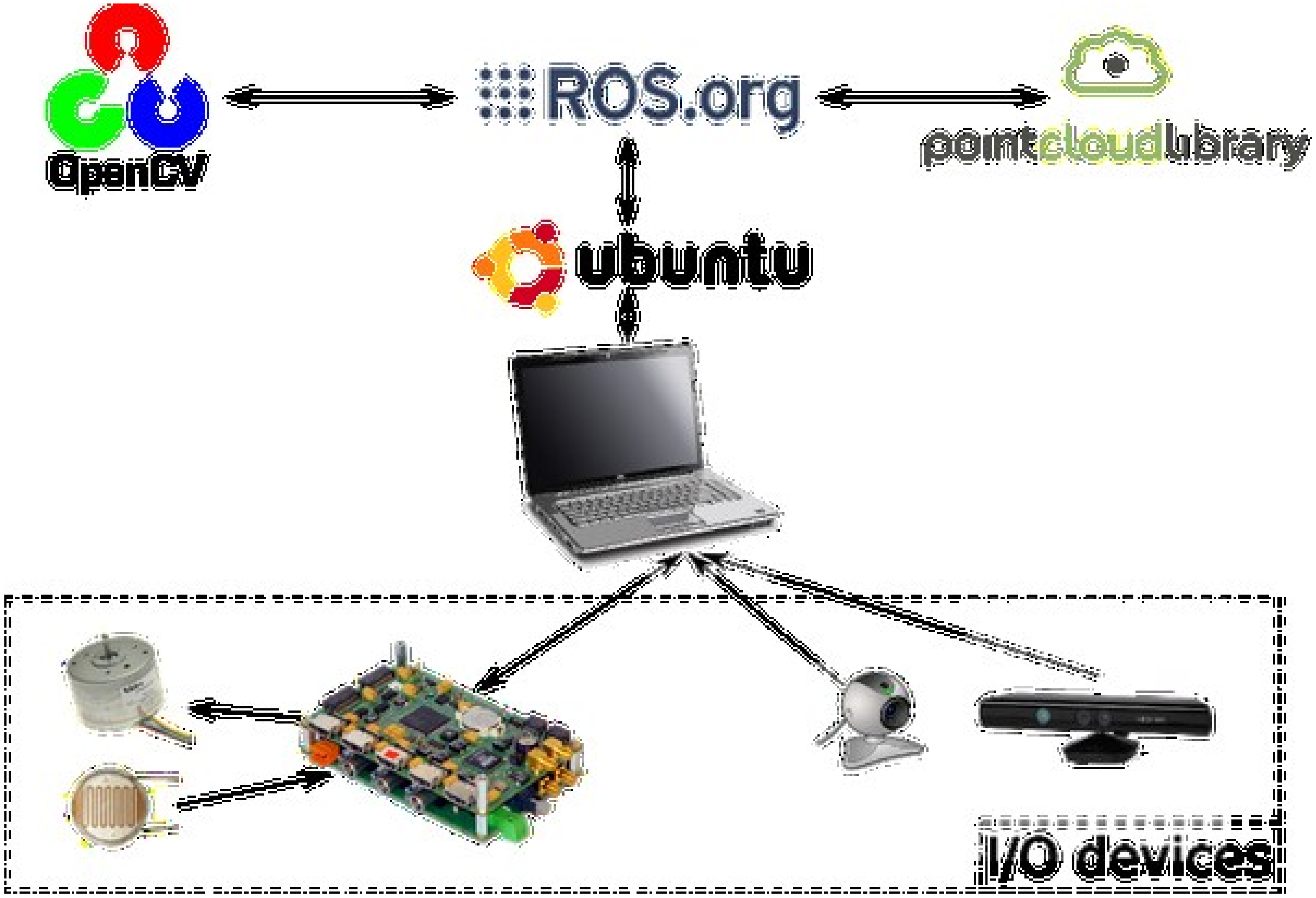
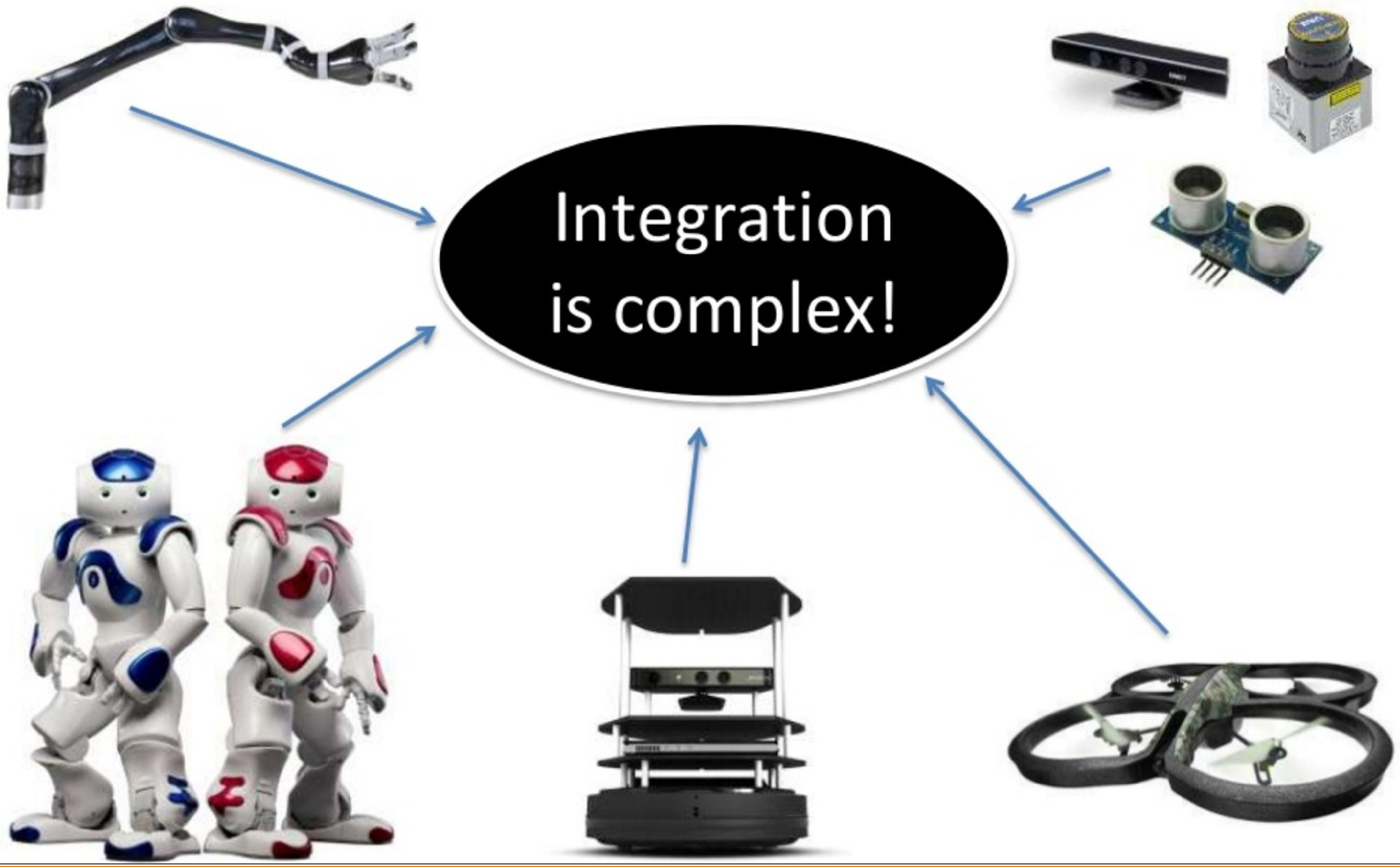
EE565-Lab1
Introduction to ROS



Dr. –Ing. Ahmad Kamal Nasir

20 Jan 2016

Challenge in Robotics

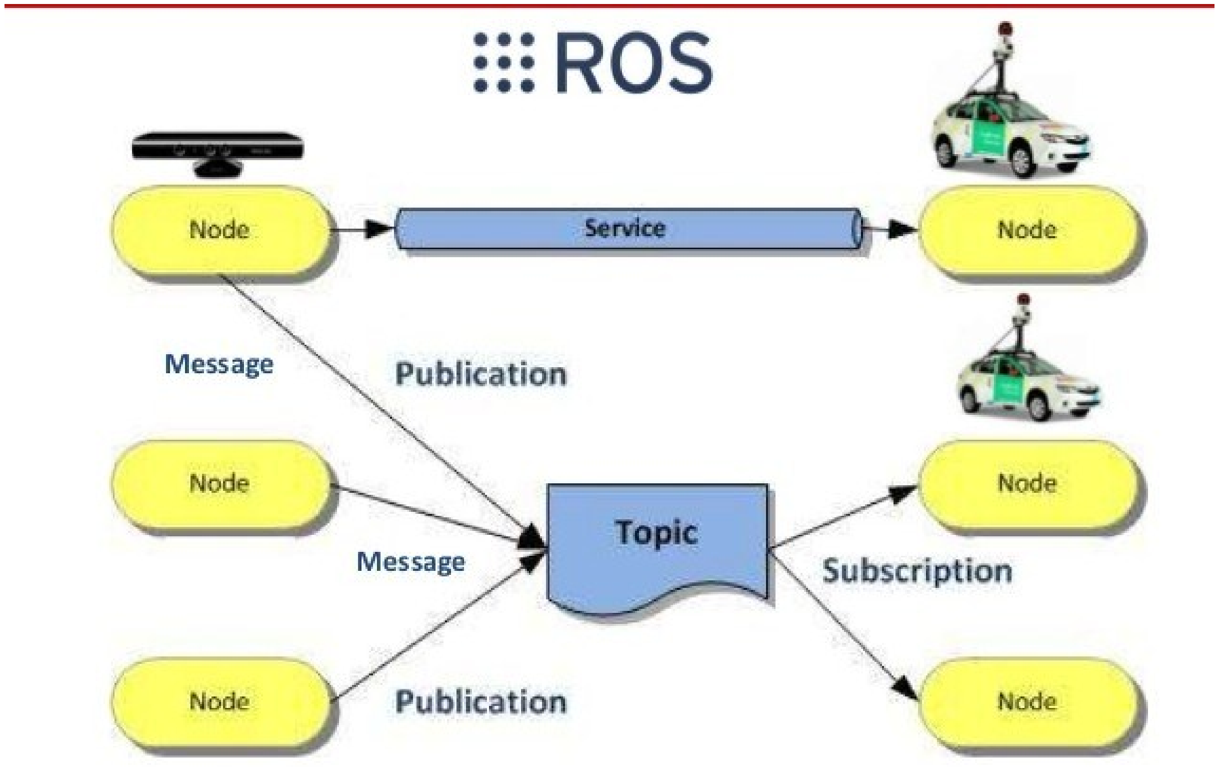


What is ROS

- stands for Robot Operating System
- Open-Source operating system
- Provides
  - Hardware abstraction
  - Low level device control
  - Message passing between processes
  - Package management.
  - C++/Python Implementation

ROS Framework

- Uses peer-to-peer network of processes
- Processing data together.
- Main components
  - Nodes
  - Master
  - Topics
  - Messages
  - Services
  - Bags

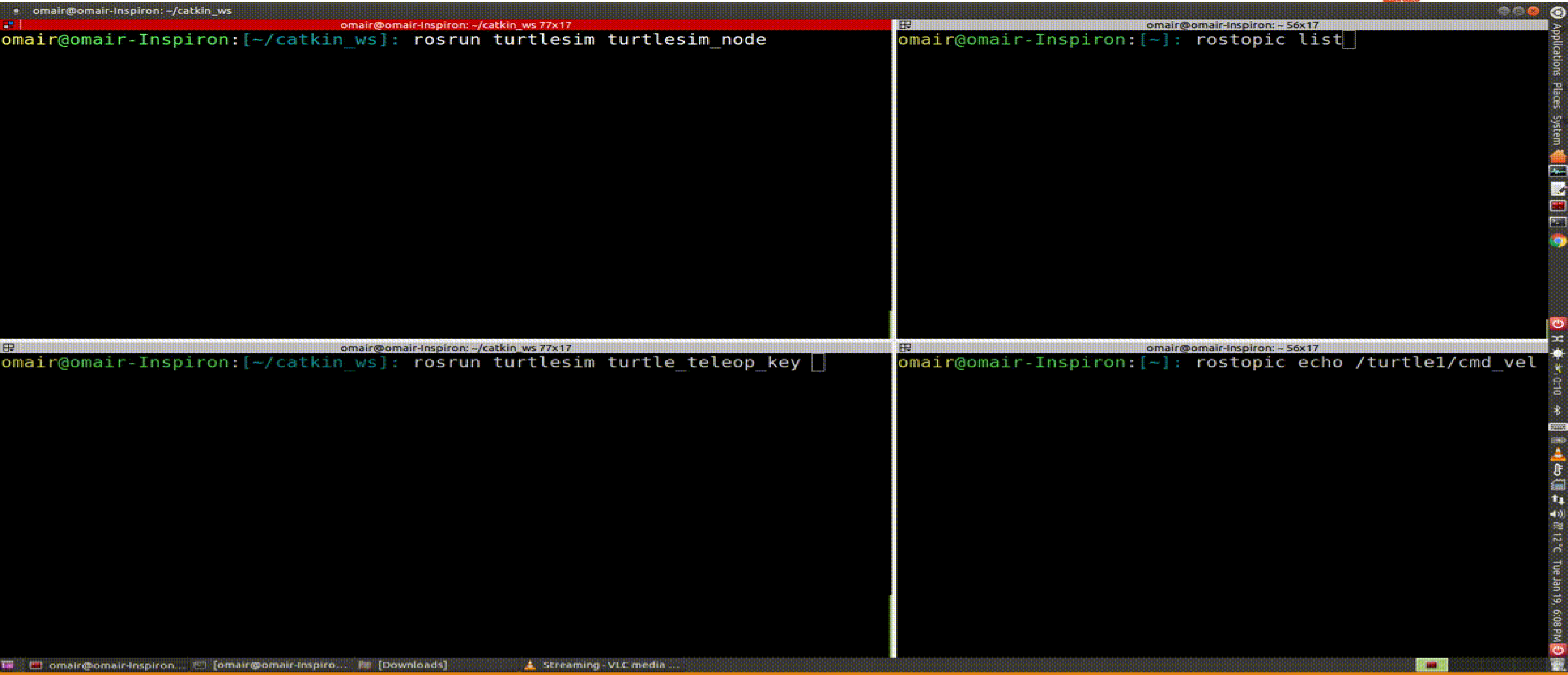


1. Nodes

- Performs computation
- Communicate with each other using
  - Topics
  - Services
  - Server
- Command: rosnode

Example: Turtle Bot

- Example
  - 1st node run Turtle Robot
  - 2nd controls robot wheel motions
  - 3rd gives graphical view of robot's pose



```
omair@omair-Inspiron:[~/catkin_ws]: rosrun turtlesim turtlesim_node
```

```
omair@omair-Inspiron:[~]: rostopic list
```

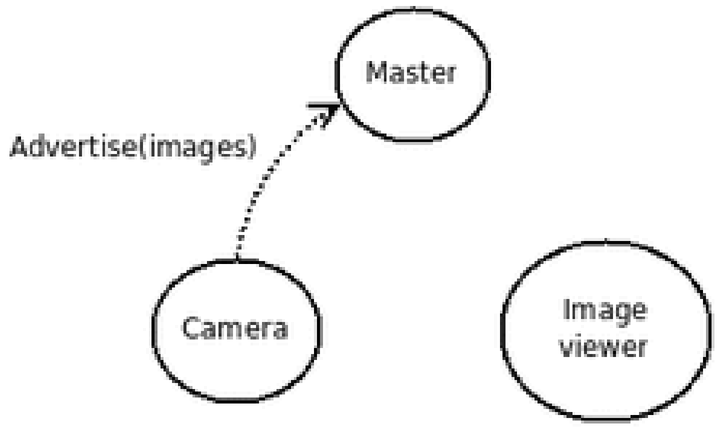
```
omair@omair-Inspiron:[~/catkin_ws]: rosrun turtlesim turtle_teleop_key
```

```
omair@omair-Inspiron:[~]: rostopic echo /turtle1/cmd_vel
```

2. Master

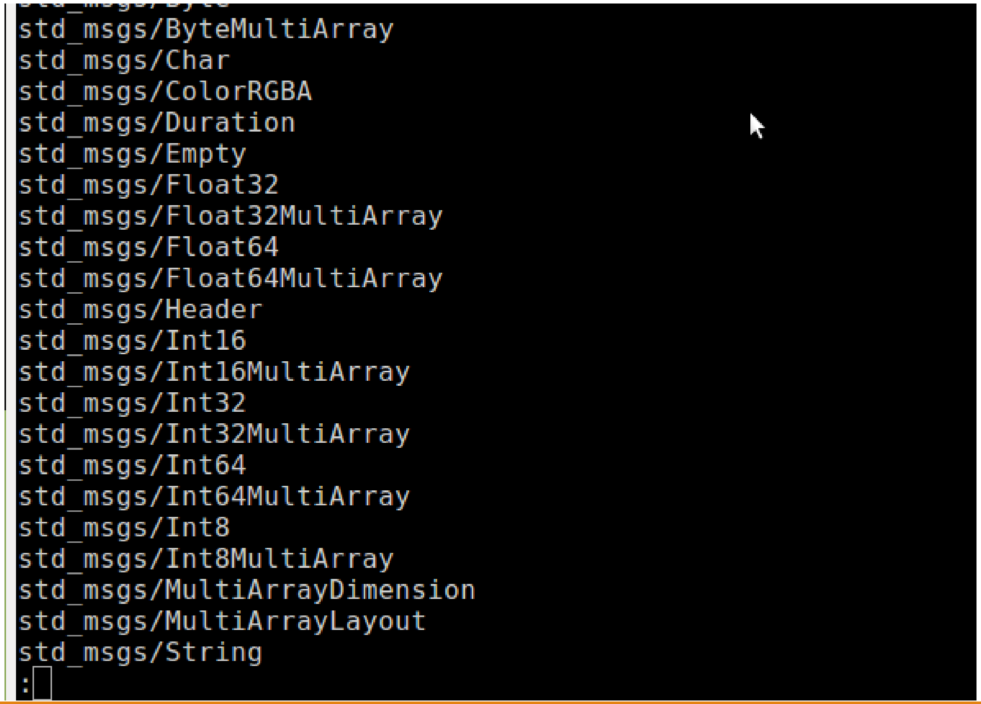
- Provides naming/registration services to nodes
- Keeps track of publishers/subscribers to

  topics as well as services.
- Enable ROS nodes to locate one another
- Once located, nodes communicate with each other peer-to-peer



3. Messages

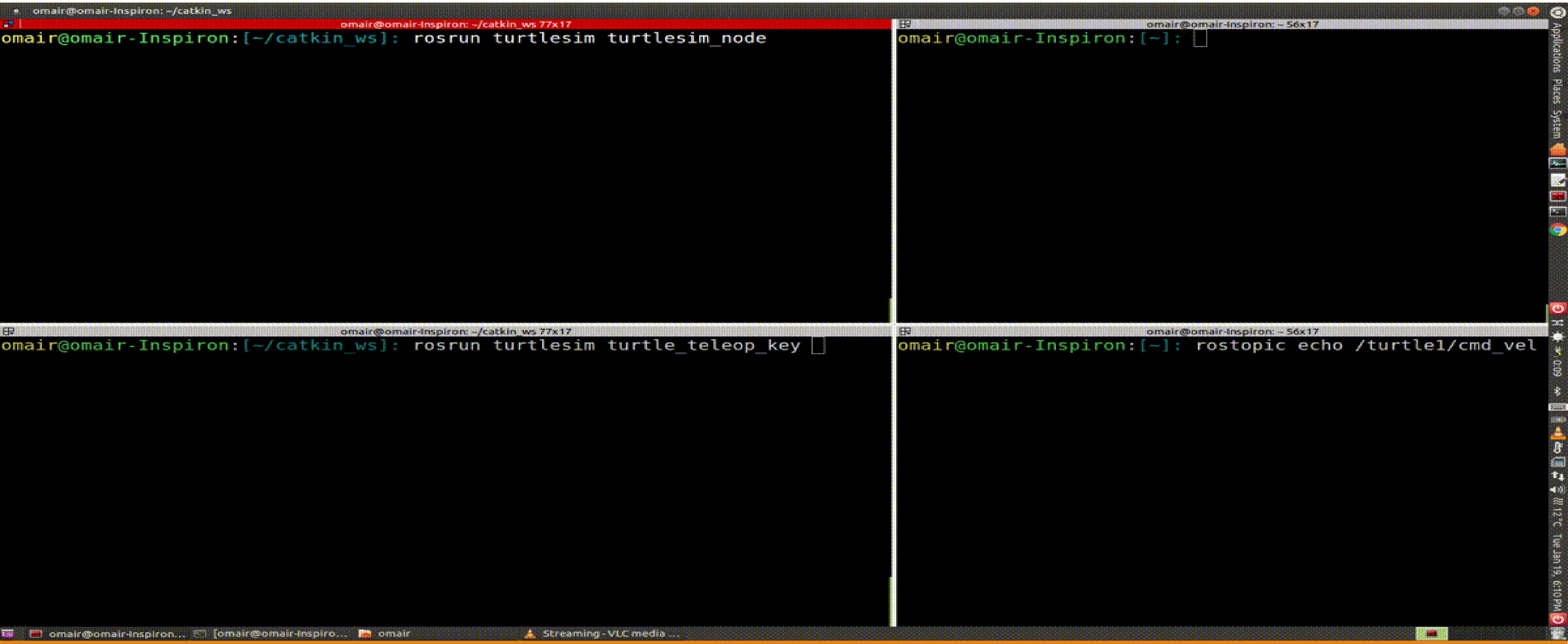
- Nodes communicate with eachother by publishing messages to topics
- Message is simple data structure comprising of typed fields.
- Data structure of message is stored in sub-directory “msg” of a package.
- Example std_msgs/msg/String.msg has message type std_msgs/String
- Command: rosmsg
  - Displays information about messages



```
std_msgs/ByteMultiArray
std_msgs/Char
std_msgs/ColorRGBA
std_msgs/Duration
std_msgs/Empty
std_msgs/Float32
std_msgs/Float32MultiArray
std_msgs/Float64
std_msgs/Float64MultiArray
std_msgs/Header
std_msgs/Int16
std_msgs/Int16MultiArray
std_msgs/Int32
std_msgs/Int32MultiArray
std_msgs/Int64
std_msgs/Int64MultiArray
std_msgs/Int8
std_msgs/Int8MultiArray
std_msgs/MultiArrayDimension
std_msgs/MultiArrayLayout
std_msgs/String
:
```

4. Topics

- Named buses over which nodes exchange messages.
- Have publish/subscribe semantics.
- Nodes subscribe to a relevant topic to get data
- Nodes publish data to relevant topic to generate data.


omair@omair-Inspiron: ~/catkin_ws
omair@omair-Inspiron: ~/catkin_ws 77x17
omair@omair-Inspiron: ~ 56x17
omair@omair-Inspiron: ~/catkin_ws 77x17
omair@omair-Inspiron: ~ 56x17
Applications Places System
omair@omair-Inspiron...
[omair@omair-Inspiro...
omair
Streaming - VLC media ...

```
omair@omair-Inspiron:[~/catkin_ws]: rosrun turtlesim turtlesim_node
```

```
omair@omair-Inspiron:[~]: 
```

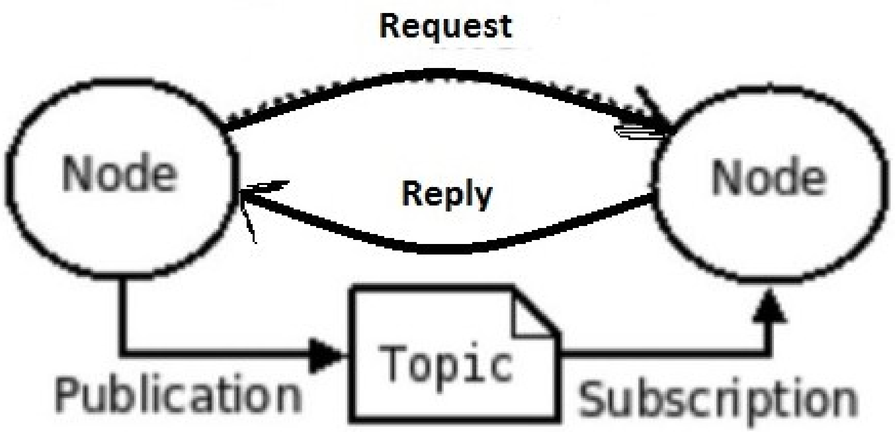
```
omair@omair-Inspiron:[~/catkin_ws]: rosrun turtlesim turtle_teleop_key 
```

```
omair@omair-Inspiron:[~]: rostopic echo /turtle1/cmd_vel
```

DR. –ING. AHMAD KAMAL NASIR
20 JAN 2016

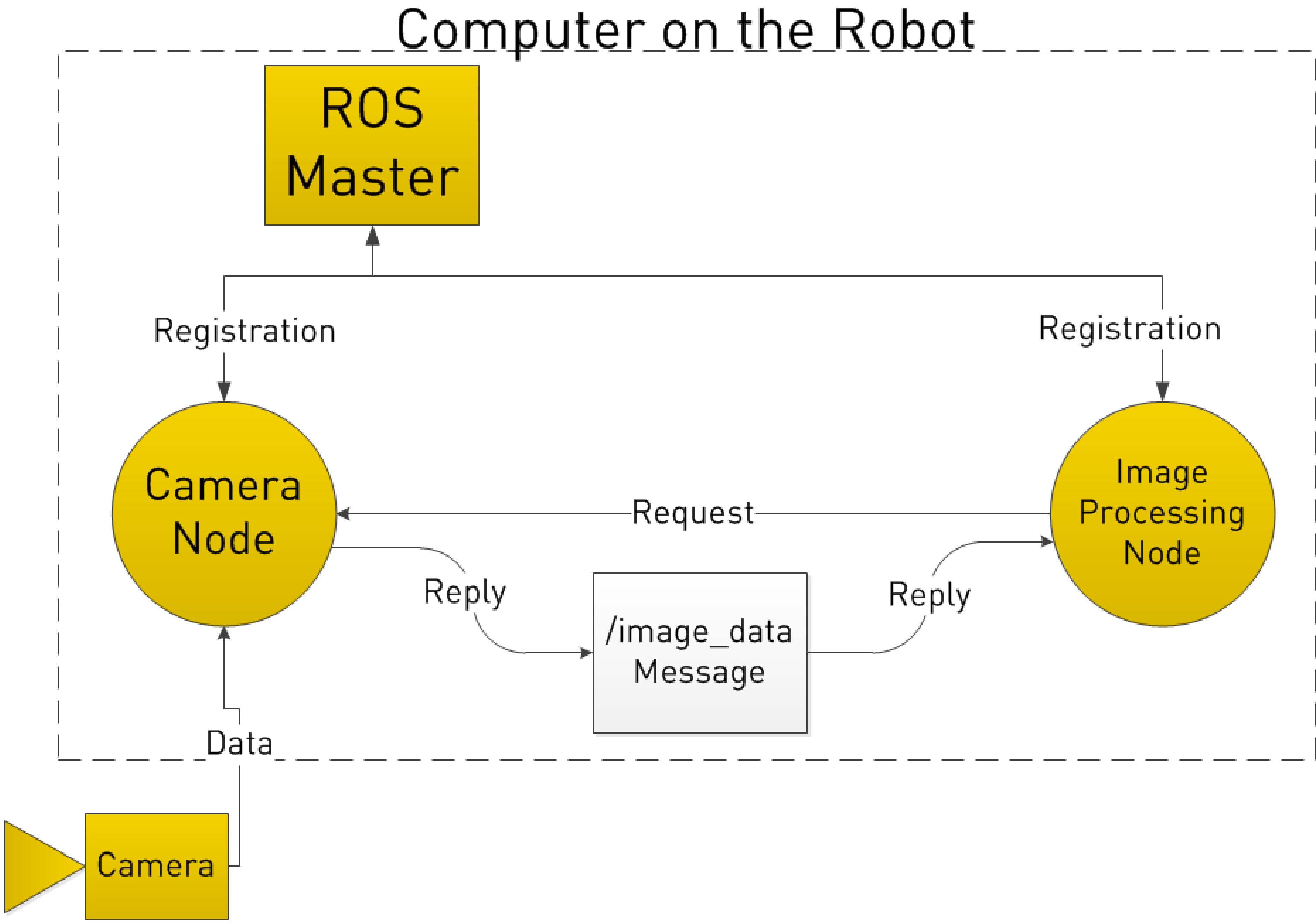
5. Services

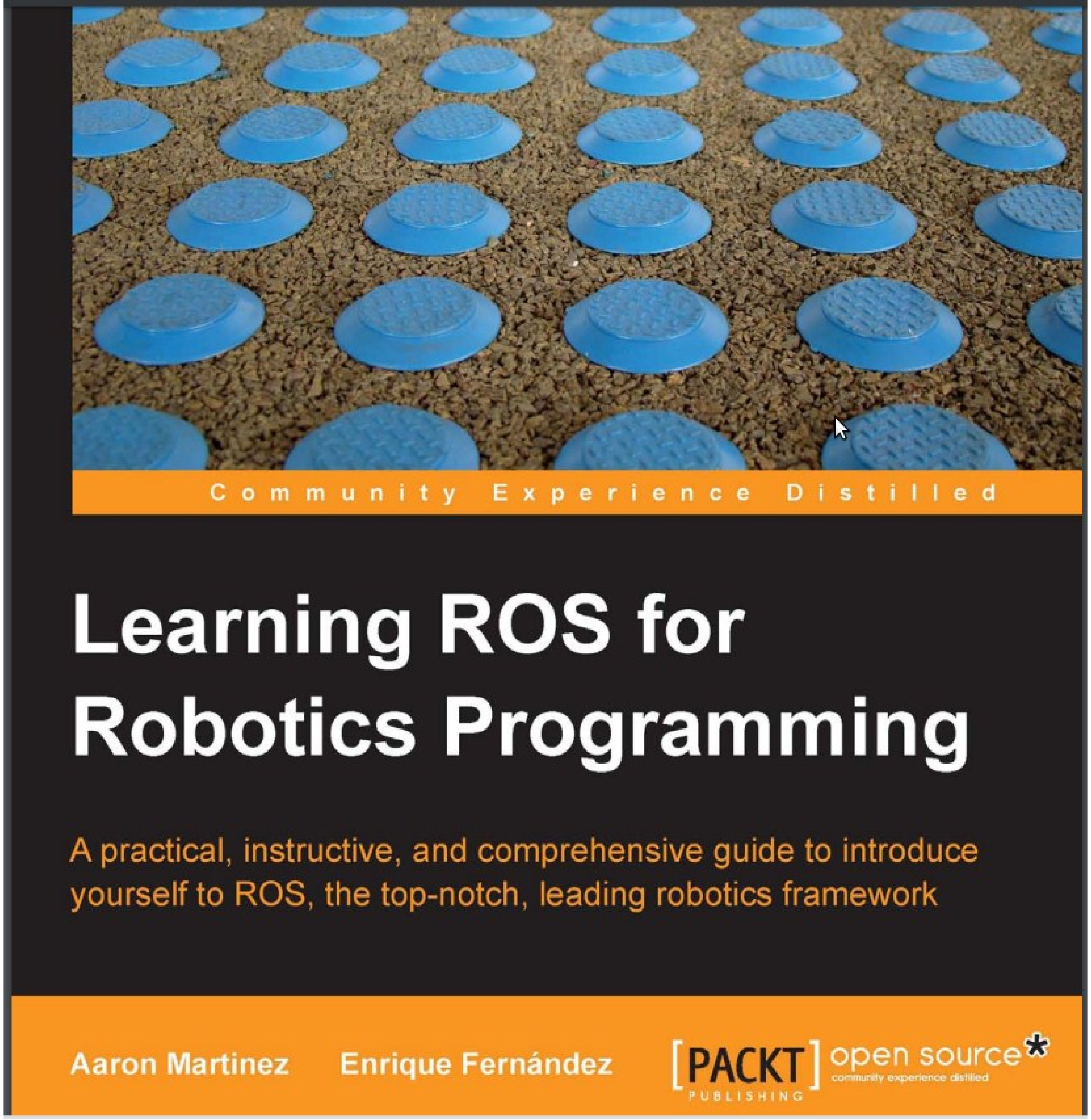
- Request/Reply within nodes is doing using a service.
- Pair of messages
  - One for request
  - One for reply
- Node offers a service under string name and client calls service by sending request message and awaiting reply.



6. Bag

- File format used to store ROS message data
- Subscribes to one or more topics and store message data as it is received.
- Played back in ROS to generate same data on topics.
- Offline use and data migration


Computer on the Robot
ROS Master
Registration
Registration
Camera Node
Request
Image Processing Node
Reply
/image_data Message
Reply
Data
Camera



Community Experience Distilled

Learning ROS for Robotics Programming

A practical, instructive, and comprehensive guide to introduce yourself to ROS, the top-notch, leading robotics framework

Aaron Martinez Enrique Fernández

[PACKT] open source*
community experience distilled
PUBLISHING

In-Lab Task

- Go to wiki.ros.org/ROS/Tutorials
- Start with Beginner Level
  1. Installing and Configuring your ROS Environment
  2. Type these 2 commands in terminal
     - rosrun turtlesim turtlesim_node
     - rosrun turtlesim turtle_teleop_key
  3. Navigating ROS Filesystem
  4. Creating ROS Package
  5. Building ROS Package
  6. Understanding ROS Nodes
  7. Understanding ROS Topics
  8. Creating ROS msg & srv
  9. Writing Publisher/Subscriber
  10. Writing Service & Client