

Sistemas Autônomos

através do Robot Operating System (ROS)

Prof. André Schneider de Oliveira
Prof. João Alberto Fabro

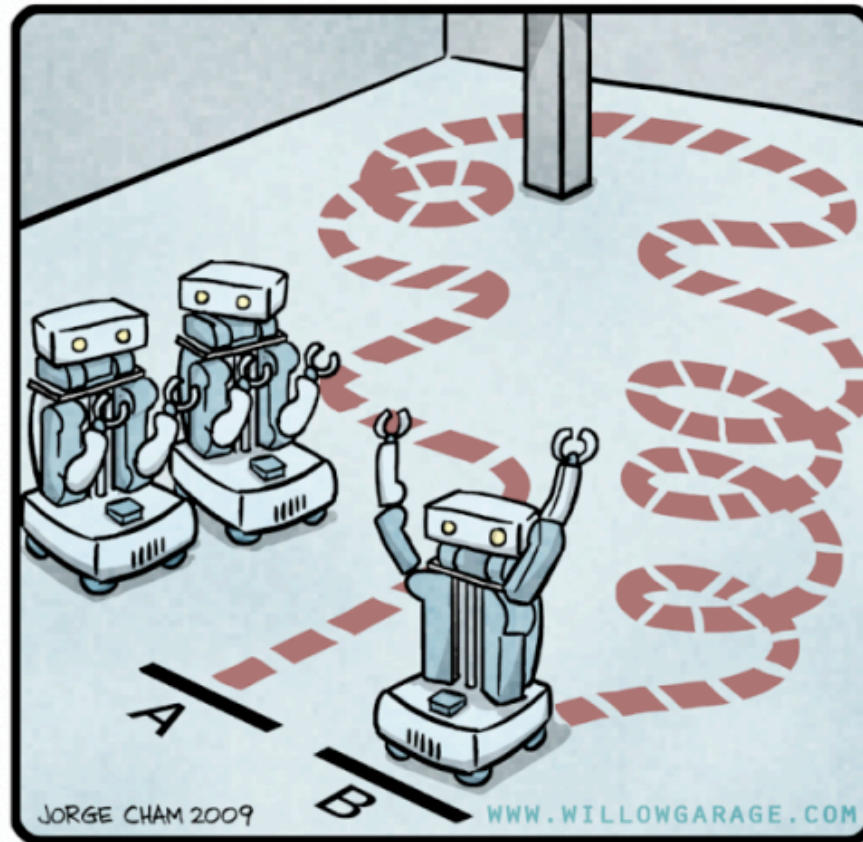
Universidade Tecnológica Federal do Paraná (UTFPR)

Robótica



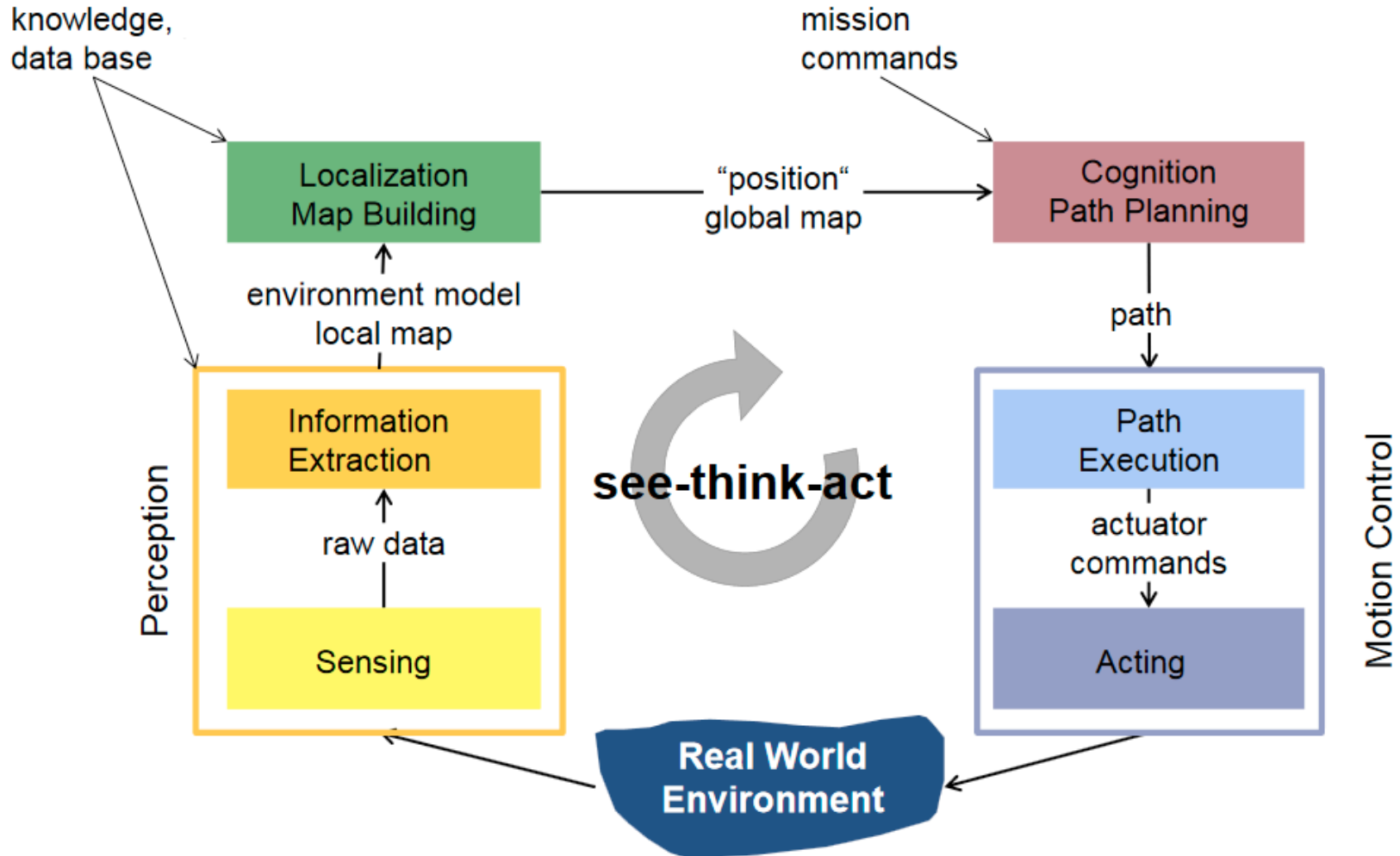
Autonomia

R.O.B.O.T. Comics



"HIS PATH-PLANNING MAY BE
SUB-OPTIMAL, BUT IT'S GOT FLAIR."

see-think-act



Robot Operating System (ROS)

- *Framework* que atua sobre o Linux para a padronização de mensagens em sistemas robóticos

www.ros.org

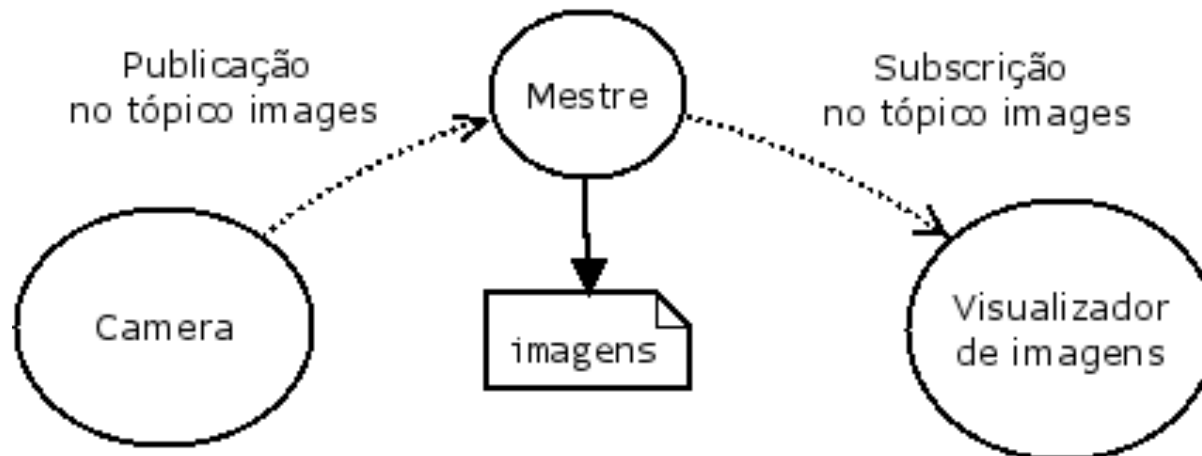
- A padronização permite o compartilhamento de soluções
- Existem diferentes versões do framework, cada uma voltada à uma versão do Linux Ubuntu
 - ROS Indigo Igloo → Ubuntu 14.04.* LTS → abril de 2019
 - ROS Jade Turtle → Ubuntu 15.04
 - ROS Kinetic Kame → Ubuntu 16.04.* LTS → abril de 2021

Robot Operating System (ROS)

- Possui nós para diferentes finalidades
 - interface com sensores e atuadores
 - comunicação entre dispositivos
 - navegação e mapeamento
 - mapeamento tridimensional
 - experimentação virtual
 - entre outros....

Robot Operating System (ROS)

- O ROS cria uma estrutura de comunicação entre nós (softwares) de diferentes origens e finalidades
- A estrutura do ROS é denominada de mestre (*master*) ROS



Master ROS

- Inicializado pelo comando

\$ roscore

```
turtlebot@turtlebot-X200CA:~$ roscore
... logging to /home/turtlebot/.ros/log/6ef6185c-9127-11e4-83da-0c84dc11754b/ros
launch-turtlebot-X200CA-9168.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://192.168.0.8:45853/
ros_comm version 1.11.9

SUMMARY
=====

PARAMETERS
* /rostdistro: indigo
* /rosversion: 1.11.9

NODES

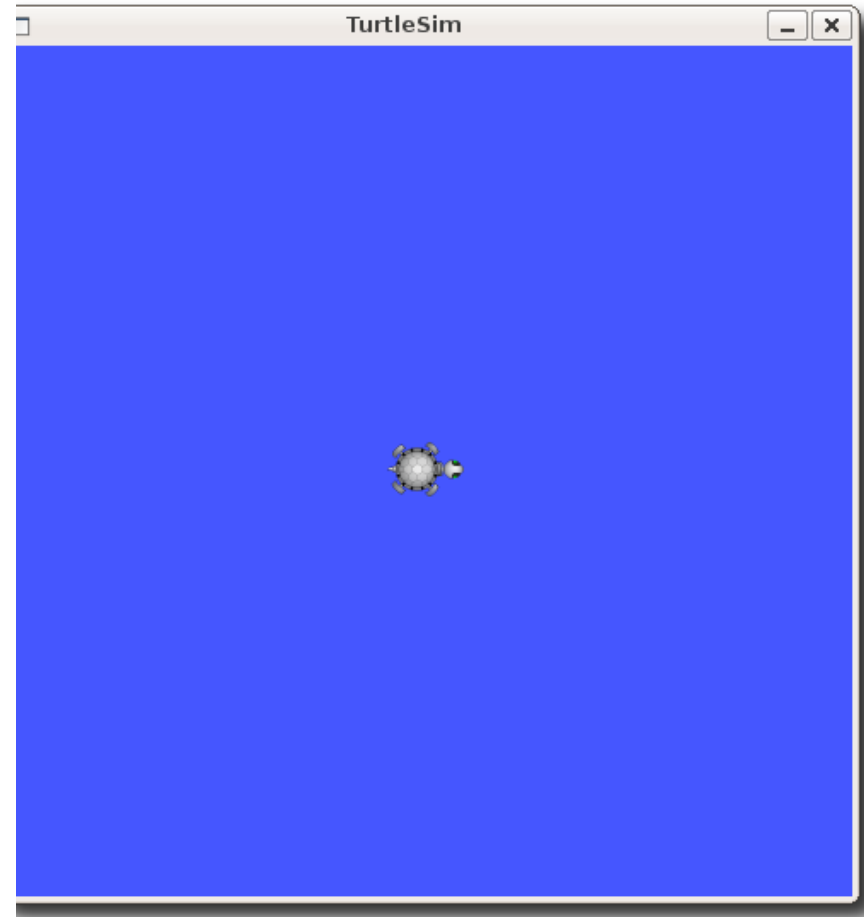
auto-starting new master
process[master]: started with pid [9180]
ROS_MASTER_URI=http://192.168.0.8:11311/

setting /run_id to 6ef6185c-9127-11e4-83da-0c84dc11754b
process[rosout-1]: started with pid [9193]
started core service [/rosout]
```


TurtleSim

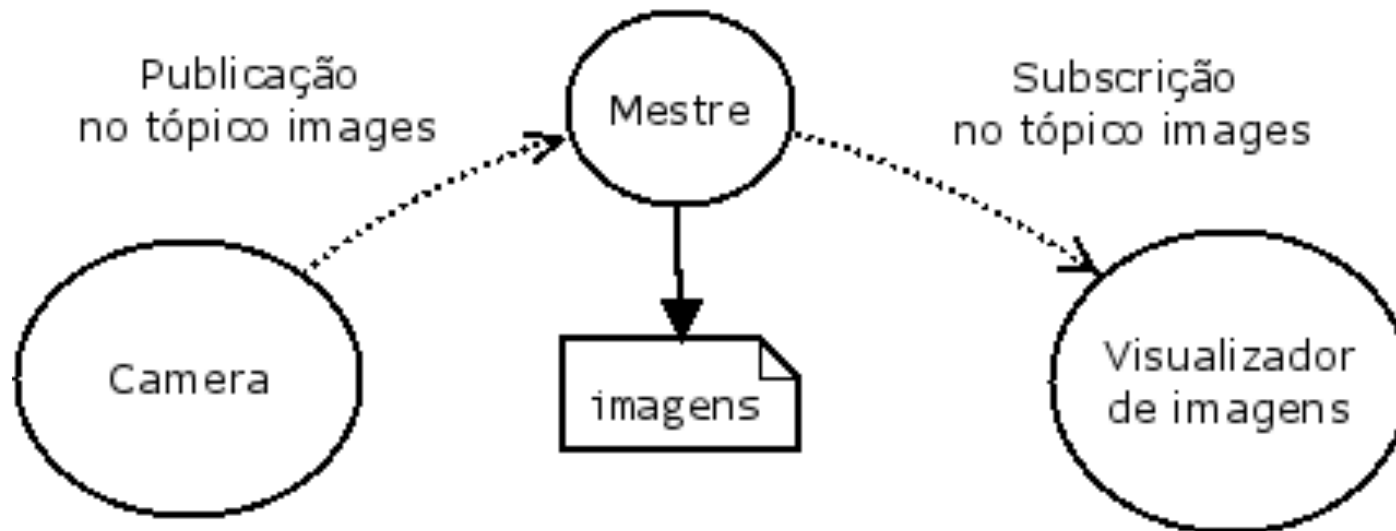
- Simulador integrado do ROS
- Executado pelo comando

```
$ roslaunch turtlesim turtlesim_node
```



Robot Operating System (ROS)

- O ROS cria uma estrutura de comunicação de softwares por lista de mensagens



ROS - Matlab

- Iniciar o nó ROS-Matlab

```
>> rosinit
```

- Finalizar o nó ROS-Matlab

```
>> roshutdown
```

Mensagens

- Existe uma grande quantidade de mensagens padronizadas no ROS
- principais bibliotecas de mensagens
 - ***std_msgs*** → mensagens primitivas → int, float, string, time
 - ***common_msgs*** → pacote com os principais tipos de mensagens
 - ***geometry_msgs*** → primitivas geométricas → acceleration, pose2D, twist
 - ***sensor_msgs*** → sensores → image, imu, pointcloud, laserscan
 - ***nav_msgs*** → navegação → gridcells, occupancygrid, path

Exemplo de Mensagens

Type: geometry_msgs/Twist

```
geometry_msgs/Vector3 linear
  float64 x
  float64 y
  float64 z
geometry_msgs/Vector3 angular
  float64 x
  float64 y
  float64 z
```

Type: nav_msgs/Odometry

```
std_msgs/Header header
  uint32 seq
  time stamp
  string frame_id
string child_frame_id
geometry_msgs/PoseWithCovariance pose
  geometry_msgs/Pose pose
    geometry_msgs/Point position
      float64 x
      float64 y
      float64 z
    geometry_msgs/Quaternion orientation
      float64 x
      float64 y
      float64 z
      float64 w
    float64[36] covariance
  geometry_msgs/TwistWithCovariance twist
    geometry_msgs/Twist twist
      geometry_msgs/Vector3 linear
        float64 x
        float64 y
        float64 z
      geometry_msgs/Vector3 angular
        float64 x
        float64 y
        float64 z
      float64[36] covariance
```

Mensagens

- Listar mensagens disponíveis

```
>> rosmg list
```

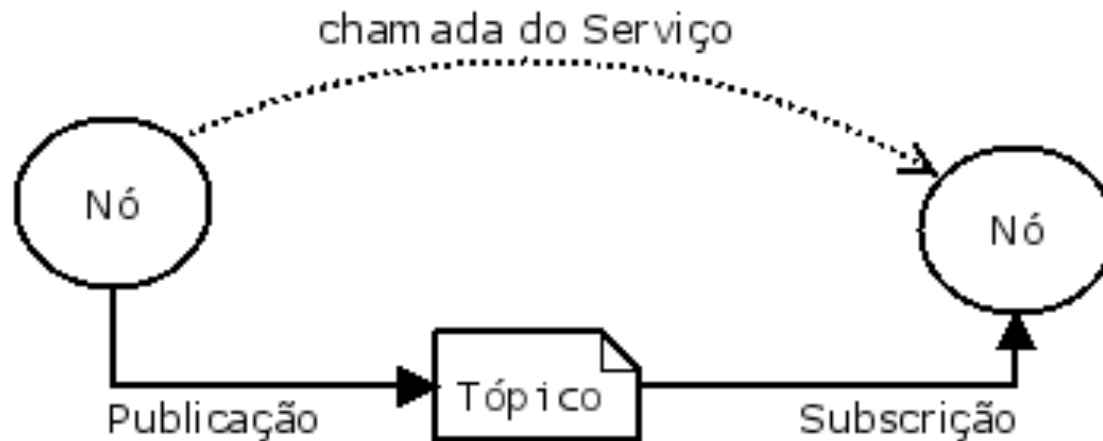
- Ver o conteúdo de uma mensagem

```
>> rosmg show [tipo_da_msg]
```

```
>> rosmg show geometry_msgs/Twist
```

Tópicos

- A lista de mensagens do ROS é subdividida em tópicos, ou espaços para mensagens
- Os tópicos armazenam as mensagens que estão trafegando pelo mestre ROS ("buffer")



Tópicos

- Os tópicos ativos do ROS podem ser visualizados por

```
>> rostopic list
```

- O conteúdo do tópico é visto por

```
>> rostopic echo [nome_do_topico]
```

```
>> rostopic echo /turtle1/pose
```

- As informações do tópico (tipo de msg) pode ser acessada pelo comando

```
>> rostopic info [nome_do_topico]
```

```
>> rostopic info /turtle1/pose
```


Nós

- Os Nós são os programas ativos no ROS que subscrevem e publicam no tópicos
- Visualizar os nós ativos

```
>> rosnode list
```

resumo ROS-Matlab

1. Interface ROS-Matlab

- **inicializar** – roscinit
- **finalizar** - rosshutdown
- **ver nós ativos** - rosnode list

2. Tópicos

- **listar tópicos ativos** - rostopic list
- **Informações sobre tópicos** - rostopic info /turtle1/pose

3. Mensagens

- **Listar tipos de mensagens** - rosmg list
- **Informações sobre mensagens** - rosmg show geometry_msgs/Twist

Referências

- Instalação do ROS Indigo

<http://wiki.ros.org/indigo/Installation/Ubuntu>

- Tutoriais do ROS-Matlab

<https://www.mathworks.com/help/robotics/robot-operating-system-ros.html>