Robot Operating System (ROS)

Walter Fetter Lages

fetter@ece.ufrgs.br

Universidade Federal do Rio Grande do Sul

Escola de Engenharia

Departamento de Sistemas Elétricos de Automação e Energia

ENG04479 Robótica A

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Introdução

ROS é um pseudo sistema operacional com bibliotecas e ferramentas para desenvolvimento de *software* para robôs:

- Gerenciamento de pacotes
- Abstração de *hardware*
- Bibliotecas com algoritmos comumente utilizados
- Simuladores
- Mecanismos de comunicação
- *scripts* úteis

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Porque ROS?

- Código aberto
- Centralização das informações
- Reuso de código
- Desenvolvimento em grupo
- Processamento inerentemente distribuído
- Nodos fracamente acoplados

Histórico

- Sistema desenvolvido em Stanford em 2000 para o robô STAIR 1
- Aperfeiçoado em 2007 pela Willow Garage para o robô PR2 e denominado ROS

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Versões do ROS

- Jade Turtle - maio de 2015
- Indigo Igloo - maio de 2014
- Hydro Medusa - setembro de 2013
- Groovy Galapagos - dezembro de 2012
- Fuerte Turtle - abril de 2012
- Electric Emys - agosto de 201;
- Diamondback - março de 2011
- C Turtle - agosto de 2010
- Box Turtle - março de 2010

Sistema Operacional *Host*

- Linux é o sistema operacional *host*
- A distribuição "oficial" é a Ubuntu
- Outras distribuição suportadas:
  - Ubuntu ARM;
  - OS X
  - OpenEmbedded/Yocto
  - Debian
  - Arch Linux
  - Ângström
  - UDOO
  - Android
  - Código fonte
  - robotpkg

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Conceitos

- O ROS é organizado utilizando o conceito de pacotes
  - Nodos
  - Bibliotecas
  - Definições de mensagens
  - Arquivos de configuração
  - *Plugins*
- Pacotes podem ser agrupados em metapacotes

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Pacotes

- Menor nível na organização do ROS
- Dedicados a uma única funcionalidade
- Cada pacote deve ser implementado em um diretório

Estrutura de um Pacote

```
nome_do_pacote
├─ src/ ..........................Códigos-fonte
├─ bin/ ..........................Executáveis
├─ lib/ ..........................Bibliotecas
├─ build/ ........................Temporários
├─ launch/ .......................Scripts de carga
├─ include/ ......Arquivos de cabeçalho
├─ package.xml ....Metadados e dependências
└─ CMakelist.txt ...Configuração do Cmake
```

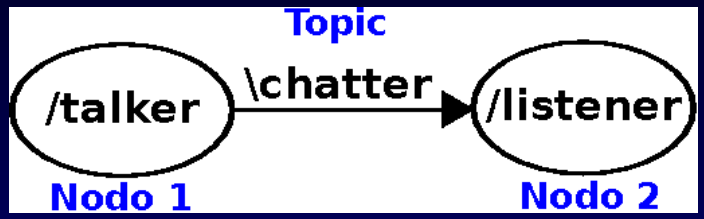
UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Metapacotes

- Agrupam pacotes que em conjunto oferecem uma funcionalidade mais abstrata
  - Pacotes não podem depender de metapacotes

Gráfico de Computação

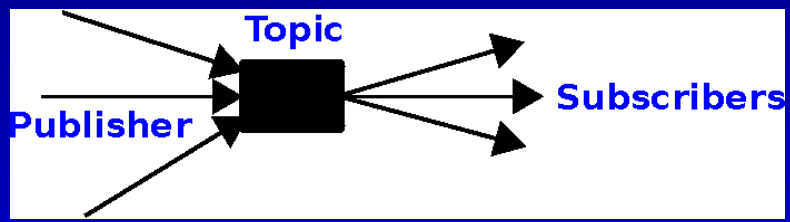
- Representa a comunicação entre os nodos



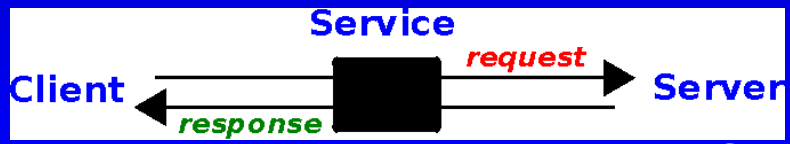
Nodo: processo do sistema operacional

Tópico: mecanismo de comunicação entre nodos do tipo *publisher/subscriber*

Mensagem: dados publicados nos tópicos



Serviço: mecanismo de comunicação entre nodos do tipo RPC



UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Mestre

- Nodo que provê serviços de registro e consulta de nomes de nodos, tópicos e serviços
- Ao iniciar, os nodos devem registrar-se com o mestre
- Ao subscrever um tópico, os nodos consultam o mestre e estabelecem conexão diretamente entre si

Servidor de Parâmetros

- Dicionário acessível aos nodos
- Parte do ROS master
- Nodos podem usar o servidor de parâmetros para armazenar e recuperar parâmetros
- Não projetado para alto desempenho
  - Adequado para parâmetros de configuração

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Tópicos

- Nodos podem publicar mensagens em tópicos
- Cada tópico pode ter vários publicadores e assinantes
- Cada nodo pode publicar ou assinar vários tópicos
- Publicadores e assinantes não sabem da existência um dos outros
- A ordem de execução não é garantida
- Comunicação assíncrona

Serviços

- Tópicos não são apropriados para solicitação de serviços entre nodos
- Serviços oferecem um mecanismo de requisição/resposta

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Action Servers

- Apropriados para serviços exigem longo tempo de execução
- Permitem o cancelamento da requisição
- Permitem receber informações sobre o *status* da execução

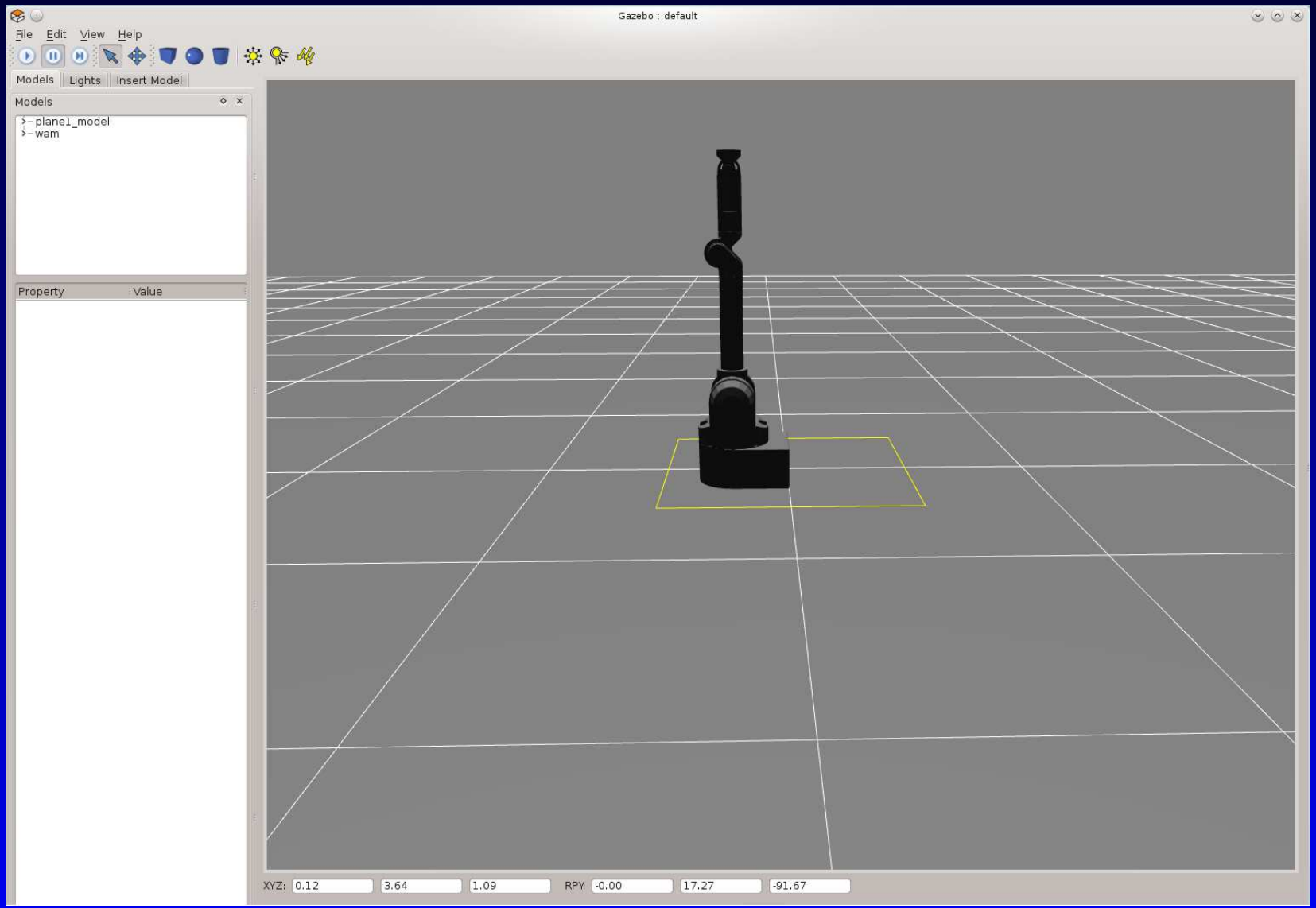
UFRGS
UNIVERSIDADE FEDERAL
DO RIO GRANDE DO SUL

Gazebo

- Simulador 3D
- Suporta diversos *backends*
  - *Open Dynamics Engine* (ODE)
  - *Bullet*
- O modelo do robô é descrito em URDF

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Barrett WAM no Gazebo



URDF

- Formato XML para descrição de robôs
- Descreve a geometria e propriedades de massa
- `<robot name="twil" >`: definição do nome do robô
- `<link name="chassis" >`: define um elo do robô
- `<mass value="6.4923"/>`: define a messa do ele
- `<inertia= ... >`: define a inércia do elo
- `<geometry> <mesh filename="package://twil/meshes/chassis.STL </geometry>`: STL com a geometria do elo
- `<joint name="right_wheel_suport_joint"type="fixed" >`: define uma junta fixa

Exemplo de URDF

```
<robot name="twil">
  <link name="chassis">
    <inertial>
      <origin xyz="6.3955E−06 −2.1963E−17 0.27338" rpy="0 0 0" />
      <mass value="6.4923" />
      <inertia ixx="0.67525" ixy="0.0014553" ixz="−0.00017525"
               iyy="0.69058" iyz="−6.3289E−18" izz="0.28611" />
    </inertial>
    <visual>
      <origin xyz="0 0 0" rpy="0 0 0" />
      <geometry>
        <mesh filename="package://twil/meshes/chassis.STL" />
      </geometry>
    </visual>
```

Exemplo de URDF

```
  <collision>
    <origin xyz="0 0 0" rpy="0 0 0" />
    <geometry>
      <mesh filename="package://twil/meshes/chassis.STL" />
    </geometry>
  </collision>
</link>
.
.
<joint name="right_wheel_suport_joint" type="fixed">
  <origin xyz="0 −0.161 −0.002" rpy="0 0 0" />
  <parent link="chassis" />
  <child link="right_wheel_suport" />
  <axis xyz="0 0 0" />
</joint>
</robot>
```

Meta-pacote `ufrgs_wam`

- Pacote para o robô Barrett WAM adaptado para o robô da UFRGS

```
ufrgs_wam/
├── CMakelist.txt
├── package.xml
├── wam_description/
└── wam_controllers/
```

Pacote wam_description

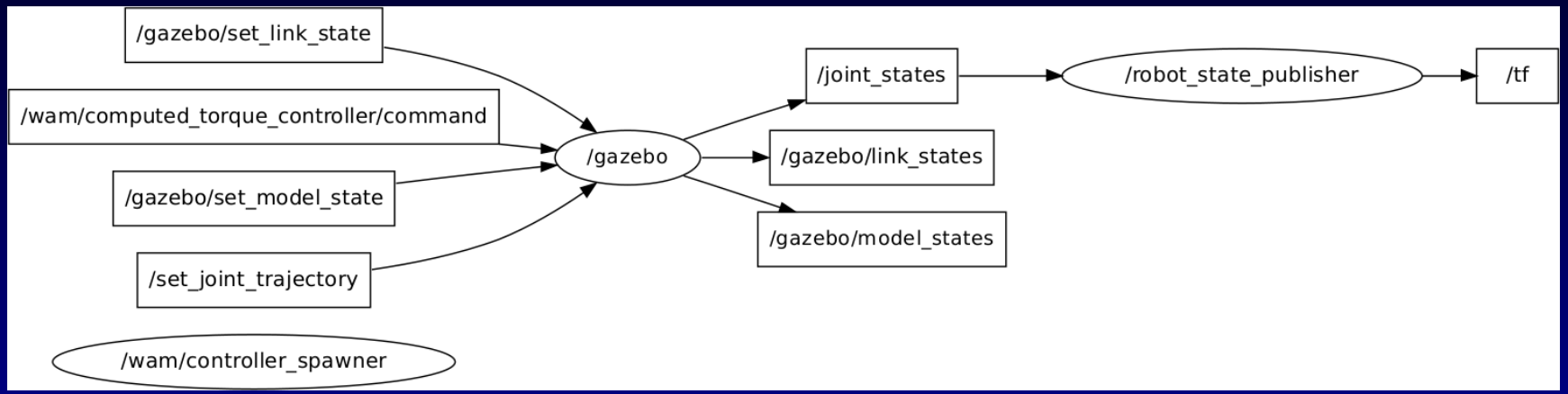
```
wam_description
├── launch/
│   ├── wam.launch
│   └── wam_sim.launch
├── package.xml
├── meshes/
│   ├── wam1.stl
│   ├── ⋮
│   └── wambase.stl
└── xacro/
    ├── wam_base.urdf.xacro
    ├── ⋮
    └── wam.urdf.xacro
```

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Pacote wam_controllers

```
wam_controllers/
├── CMakeLists.txt
├── config/
│   └── computed_torque_control.yaml
├── include/
│   └── wam_controllers/
│       └── computed_torque_controller.h
├── launch/
│   └── computed_torque.launch
├── lib/
│   └── libwam_controllers.so
├── package.xml
├── src/
│   └── computed_torque_controller.cpp
├── wam_controllers_plugins.xml
└── scripts
    └── move_home.sh
```

Gráfico de Computação do WAM



UFRGS
UNIVERSIDADE FEDERAL
DO RIO GRANDE DO SUL

Barrett WAM no Gazebo