GAZEBO

David Vargas Frutos

david.vargas@upm.es – dvargasfr@gmail.com

Centre for Automation and Robotics

Centro de Automática y Robótica

Madrid, Spain

Madrid, Octubre 2017

Antes de comenzar...

Descarga el material

http://bit.ly/cursoROS3pdf

http://bit.ly/chuletagazeboROS

Contenidos

1. Cheat Sheet
2. Breve historia
3. Gazebo Tutorials
4. Get Started
5. Build a World
6. Build a Robot
7. Connect to ROS

Cheat Sheet

MUNDOS

/usr/share/gazebo-7/worlds

IMÁGENES DE TEXTURAS

/usr/share/gazebo-7/media/materials/textures

SCRIPT DE TEXTURAS

/usr/share/gazebo-7/media/materials/scripts/gazebo.material

BASE DE DATOS LOCAL DE MODELOS

~/.gazebo/models

GAZEBO DESDE ROS

/opt/ros/kinetic/share/gazebo_ros/launch/empty_world.launch

GAZEBO DESDE CATKIN

~/catkin_ws/src/my_gazebo_package/launch/mygazebo.launch

~/catkin_ws/src/my_gazebo_package/worlds/mygazebo.world

Breve historia

History

Gazebo development began in the fall of 2002 at the University of Southern California. The original creators were Dr. Andrew Howard and his student Nate Koenig. The concept of a high-fidelity simulator stemmed from the need to simulate robots in outdoor environments under various conditions. As a complementary simulator to Stage, the name Gazebo was chosen as the closest structure to an outdoor stage. The name has stuck despite the fact that most users of Gazebo simulate indoor environments.

Over the years, Nate continued development of Gazebo while completing his PhD. In 2009, John Hsu, a Senior Research Engineer at Willow, integrated ROS and the PR2 into Gazebo, which has since become one the primary tools used in the ROS community. A few years later in the Spring of 2011, Willow Garage started providing financial support for the development of Gazebo. In 2012, Open Source Robotics Foundation (OSRF) spun out of Willow Garage and became the steward of the Gazebo project. After significant development effort by a team of talented individuals, OSRF used Gazebo to run the Virtual Robotics Challenge, a component in the DARPA Robotics Challenge, in July of 2013.

OSRF continues development of Gazebo with support from a diverse and active community. Stay tuned for more exciting developments related to robot simulation.

Willow Garage

ROS

DARPA ROBOTICS CHALLENGE

Open Source Robotics Foundation

Tutoriales de Gazebo

http://gazebosim.org/tutorials

Gazebo Tutorials

Categorized

Stand-alone tutorials categorized by topic. Click on a topic to view the tutorials in that category.

Installation
Instructions to install Gazebo on all the plaforms suppor...

Get Started
Welcome to the Gazebo tutorials! This is a great place to...

Build a Robot
The following tutorials describe how to build and modify ...

Model Editor
The following tutorials describe how to build and modify ...

Build a World
These tutorials describe how to create an environment in ...

Tools and utilities
Gazebo ships with many tools and utilities. These tutoria...

Write a plugin
Plugins allow you to control models, sensors, world prope...

Plugins
The following tutorials describe how to use plugins that ...

Sensors
A simulated sensor analyzes the environment and produces ...

Using Math
Gazebo has a custom math library. These tutorials describ...

User input
User input can take many forms, including interactions fr...

Connect to Player
Player is a robot control framework. These tutorials desc...

Web client (Gzweb)
Gzweb is a WebGL client for Gazebo. It lets you interact ...

Connect to ROS
ROS is a robot control framework. These tutorials describ...

Transport Library
Communication between the simulation server and client to...

Physics Library
A core component of Gazebo are the physics engines. These...

Rendering Library
Gazebo's rendering library is used to generate sensor dat...

Development
Customizing Gazebo is easy thanks to extensive documentat...

DRCSim
DARPA Robotics Challenge specific tutorials.

Scripting (javascript and NodeJs)
Gazebo scripting interface: NodeJS bindings for the Gazeb...

HAPTIX
The following are tutorials designed for...

Get Started

Categorized

Stand-alone tutorials categorized by topic. Click on a topic to view the tutorials in that category.

Installation
Instructions to install Gazebo on all the plaforms suppor...

Get Started
Welcome to the Gazebo tutorials! This is a great place to...

Build a Robot
The following tutorials describe how to build and modify ...

Model Editor
The following tutorials describe how to build and modify ...

Build a World
These tutorials describe how to create an environment in ...

Tools and utilities
Gazebo ships with many tools and utilities. These tutoria...

Write a plugin
Plugins allow you to control models, sensors, world prope...

Plugins
The following tutorials describe how to use plugins that ...

Sensors
A simulated sensor analyzes the environment and produces ...

Using Math
Gazebo has a custom math library. These tutorials describ...

User input
User input can take many forms, including interactions fr...

Connect to Player
Player is a robot control framework. These tutorials desc...

Web client (Gzweb)
Gzweb is a WebGL client for Gazebo. It lets you interact ...

Connect to ROS
ROS is a robot control framework. These tutorials describ...

Transport Library
Communication between the simulation server and client to...

Physics Library
A core component of Gazebo are the physics engines. These...

Rendering Library
Gazebo's rendering library is used to generate sensor dat...

Development
Customizing Gazebo is easy thanks to extensive documentat...

DRCSim
DARPA Robotics Challenge specific tutorials.

Scripting (javascript and NodeJs)
Gazebo scripting interface: NodeJS bindings for the Gazeb...

HAPTIX
The following are tutorials designed for...

Get Started

STANDALONE

Mundos disponibles (archivos .world)

Ubicación

```
/usr/share/gazebo-7/worlds/
```

Modelos interesantes

gazebo worlds/cessna_demo.world
gazebo worlds/everything.world
gazebo worlds/mud.world
gazebo worlds/pioneer2dx.world
gazebo worlds/pr2.world
gazebo worlds/robocup_3Dsim.world
gazebo worlds/willowgarage.world

Get Started

STANDALONE

Modelos disponibles (archivos .sdf)

Ubicación

Base de datos local en `~/.gazebo/models`

Base de datos online en https://gazebosim.org/models/

Variables de entorno

Incorporar al .bashrc la instrucción que inicializa las variables de entorno

```
echo "source <install_path>/share/gazebo/setup.sh" >> ~/.bashrc
```

es decir,

```
$ echo "source /usr/share/gazebo-7/setup.sh" >> ~/.bashrc
```

Para comprobar que se han inicializado las variables de entorno

```
$ env | grep GAZEBO
```

Build a World

Categorized

Stand-alone tutorials categorized by topic. Click on a topic to view the tutorials in that category.

Installation
Instructions to install Gazebo on all the plaforms suppor...

Get Started
Welcome to the Gazebo tutorials! This is a great place to...

Build a Robot
The following tutorials describe how to build and modify ...

Model Editor
The following tutorials describe how to build and modify ...

Build a World
These tutorials describe how to create an environment in ...

Tools and utilities
Gazebo ships with many tools and utilities. These tutoria...

Write a plugin
Plugins allow you to control models, sensors, world prope...

Plugins
The following tutorials describe how to use plugins that ...

Sensors
A simulated sensor analyzes the environment and produces ...

Using Math
Gazebo has a custom math library. These tutorials describ...

User input
User input can take many forms, including interactions fr...

Connect to Player
Player is a robot control framework. These tutorials desc...

Web client (Gzweb)
Gzweb is a WebGL client for Gazebo. It lets you interact ...

Connect to ROS
ROS is a robot control framework. These tutorials describ...

Transport Library
Communication between the simulation server and client to...

Physics Library
A core component of Gazebo are the physics engines. These...

Rendering Library
Gazebo's rendering library is used to generate sensor dat...

Development
Customizing Gazebo is easy thanks to extensive documentat...

DRCSim
DARPA Robotics Challenge specific tutorials.

Scripting (javascript and NodeJs)
Gazebo scripting interface: NodeJS bindings for the Gazeb...

HAPTIX
The following are tutorials designed for...

Build a World

EDITAR LA ESCENA

Añadir modelos

Figuras simples (desde el editor)
Modelos de base de datos local/online (pestaña *Insert*)

Editar modelos

Translación
Rotación
Escalado

Eliminar modelos

Tecla *Delete*
Botón derecho – Delete

Guardar un mundo

File – Save World
File – Save World As

Build a Robot

Categorized

Stand-alone tutorials categorized by topic. Click on a topic to view the tutorials in that category.

Installation
Instructions to install Gazebo on all the plaforms suppor...

Get Started
Welcome to the Gazebo tutorials! This is a great place to...

Build a Robot
The following tutorials describe how to build and modify ...

Model Editor
The following tutorials describe how to build and modify ...

Build a World
These tutorials describe how to create an environment in ...

Tools and utilities
Gazebo ships with many tools and utilities. These tutoria...

Write a plugin
Plugins allow you to control models, sensors, world prope...

Plugins
The following tutorials describe how to use plugins that ...

Sensors
A simulated sensor analyzes the environment and produces ...

Using Math
Gazebo has a custom math library. These tutorials describ...

User input
User input can take many forms, including interactions fr...

Connect to Player
Player is a robot control framework. These tutorials desc...

Web client (Gzweb)
Gzweb is a WebGL client for Gazebo. It lets you interact ...

Connect to ROS
ROS is a robot control framework. These tutorials describ...

Transport Library
Communication between the simulation server and client to...

Physics Library
A core component of Gazebo are the physics engines. These...

Rendering Library
Gazebo's rendering library is used to generate sensor dat...

Development
Customizing Gazebo is easy thanks to extensive documentat...

DRCSim
DARPA Robotics Challenge specific tutorials.

Scripting (javascript and NodeJs)
Gazebo scripting interface: NodeJS bindings for the Gazeb...

HAPTIX
The following are tutorials designed for...

Antes de continuar...

Repositorio software

https://bitbucket.org/dvargasfr/

... CONSULTA EL CÓDIGO PROPORCIONADO ...

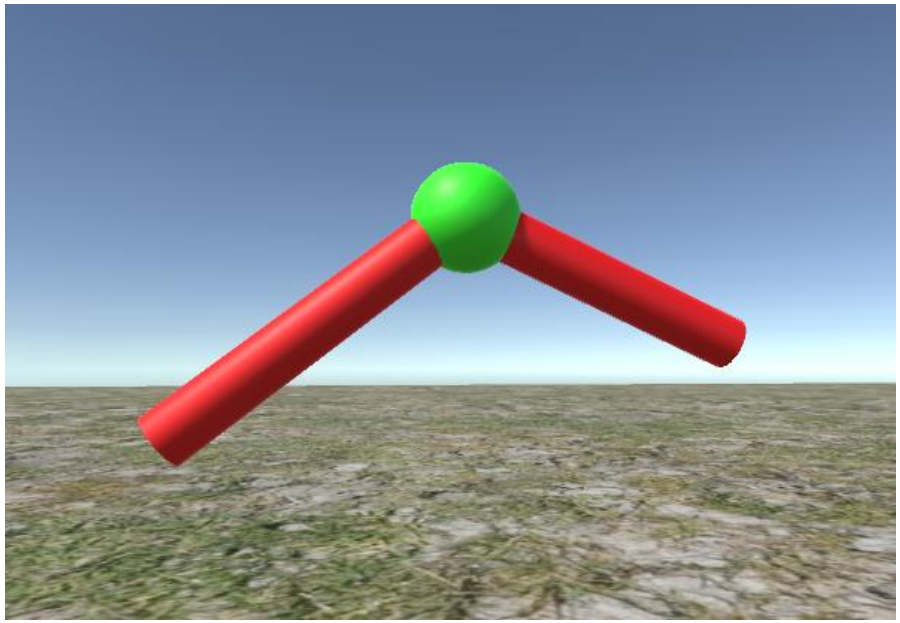
Build a Robot

COMPONENTES DE UN MODELO SDF

Links
Propiedades físicas de un objeto del modelo (Collision, Visual, Inertial, Sensor).

Joints
Conector entre dos o más Links.

Plugins
Código que realiza el control del modelo.



Made with unity

Build a Robot

ESTRUCTURA DE LOS MODELOS

model.config
Nombre, versión, autor, descripción

```
<?xml version="1.0"?>
<model>
  <name>My Robot</name>
  <version>1.0</version>
  <sdf version='1.4'>model.sdf</sdf>

  <author>
   <name>My Name</name>
   <email>me@my.email</email>
  </author>

  <description>
    My awesome robot.
  </description>
</model>
```

model.sdf
Parámetros del modelo en formato SDF (*Simulator Description Format*)

```
<?xml version='1.0'?>
<sdf version='1.4'>
  <model name="my_robot">
  </model>
</sdf>
```

Build a Robot

CREANDO UN ROBOT MÓVIL

Crear el modelo en la base de datos local

```
$ mkdir –p ~/.gazebo/models/my_robot
```

Crear el archivo de configuración del modelo

```
$ gedit ~/.gazebo/models/my_robot/model.cofig
```

```
$ gedit ~/.gazebo/models/my_robot/model.sdf
```

Build a Robot

AÑADIR TEXTURAS

Imágenes de texturas en:

```
/usr/share/gazebo-7/media/materials/textures
```

Para copiar imágenes en este directorio hay que tener permisos de admin:

```
$ sudo cp kinetic.png /usr/share/gazebo-7/media/materials/textures
```

Scripts para texturizar en:

```
/usr/share/gazebo-7/media/materials/scripts/gazebo.material
```

Crear material Kinetic (copia de Runway):

```
material Gazebo/Kinetic{
        …
        texture kinetic.png
        …
}
```

Referenciar esta textura en el model.sdf (elemento <material> dentro de <visual>)

```
<material>
    <script>
        <name>Gazebo/Kinetic</name>
    </script>
</material>
```

Connect to ROS

Categorized

Stand-alone tutorials categorized by topic. Click on a topic to view the tutorials in that category.

Installation
Instructions to install Gazebo on all the plaforms suppor...

Get Started
Welcome to the Gazebo tutorials! This is a great place to...

Build a Robot
The following tutorials describe how to build and modify ...

Model Editor
The following tutorials describe how to build and modify ...

Build a World
These tutorials describe how to create an environment in ...

Tools and utilities
Gazebo ships with many tools and utilities. These tutoria...

Write a plugin
Plugins allow you to control models, sensors, world prope...

Plugins
The following tutorials describe how to use plugins that ...

Sensors
A simulated sensor analyzes the environment and produces ...

Using Math
Gazebo has a custom math library. These tutorials describ...

User input
User input can take many forms, including interactions fr...

Connect to Player
Player is a robot control framework. These tutorials desc...

Web client (Gzweb)
Gzweb is a WebGL client for Gazebo. It lets you interact ...

Connect to ROS
ROS is a robot control framework. These tutorials describ...

Transport Library
Communication between the simulation server and client to...

Physics Library
A core component of Gazebo are the physics engines. These...

Rendering Library
Gazebo's rendering library is used to generate sensor dat...

Development
Customizing Gazebo is easy thanks to extensive documentat...

DRCSim
DARPA Robotics Challenge specific tutorials.

Scripting (javascript and NodeJs)
Gazebo scripting interface: NodeJS bindings for the Gazeb...

HAPTIX
The following are tutorials designed for...

Connect to ROS

INSTALACIÓN

```
$ sudo apt-get install ros-kinetic-gazebo-ros-pkgs ros-kinetic-gazebo-ros-control
```

Configuración del .bashrc hasta ahora:

```
source /opt/ros/kinetic/setup.bash
source ~/catkin_ws/devel/setup.bash
source /usr/share/gazebo-7/setup.sh
```

Arrancar Gazebo desde ROS

```
$ roscore  (en una terminal)
$ rosrun gazebo_ros gazebo  (en otra terminal)
```

Verificar la conexión

```
$ rostopic list
$ rosservice list
```

Connect to ROS

ROSLAUNCH

Cargar un mundo vacío de Gazebo desde ROS

```
$ roslaunch gazebo_ros empty_world.launch
```

Ubicación del fichero launch

```
/opt/ros/kinetic/share/gazebo_ros/launch/empty_world.launch
```

Por defecto carga un mundo (archivo .world) vacío, empty.world

```
<arg name="world_name" default="worlds/empty.world"/>
```

RECUERDA: La ubicación de empty.world es `/usr/share/gazebo-7/worlds/`

Podemos cargar un mundo de Gazebo descrito en alguno de estos ficheros `.world` tomando como base el launcher del mundo vacío `empty_world.launch`:

I. modificando el propio archivo `.launch` (observa cómo se carga el mundo `mud.world` en el launcher `mud_world.lauch` a partir de `empty_world.launch`)

```
/opt/ros/kinetic/share/gazebo_ros/launch/mud_world.launch
```

II. pasándolo como argumento.

```
$ roslaunch gazebo_ros empty_world.launch world_name:=worlds/empty_sky.world
```

Connect to ROS

ROSLAUNCH

Crear un mundo personalizado

Crear paquete en workspace

RECUERDA: El comando `catkin_create_pkg` debe ejecutarse desde el directorio src del workspace

```
$ cd ~/catkin_ws/src
$ catkin_create_pkg myworldgazebo std_msgs roscpp
$ cd myworldgazebo
```

Crear fichero myworldgazebo.launch dentro de un directorio launch

```
$ mkdir launch
$ touch launch/myworldgazebo.launch
```

Crear fichero myworldgazebo.world dentro de un directorio worlds

```
$ mkdir worlds
$ touch worlds/myworldgazebo.world
```

… CONSULTA EL CÓDIGO PROPORCIONADO …

Lanzar el mundo que esté aquí definido

```
$ roslaunch myworldgazebo myworldgazebo.launch
```

Connect to ROS

PLUGINS DE GAZEBO EN ROS

Para añadir un plugin a un SDF hay que incluirlo dentro del elemento `<gazebo>`
Para añadir un plugin a un URDF hay que incluirlo dentro del elemento `<model>`

Los plugins se encuentran en la ubicación referenciada por la variable de entorno `GAZEBO_PLUGIN_PATH`, esta variable se consulta mediante

```
$ echo $GAZEBO_PLUGIN_PATH
```

Cómo encontrar la ubicación de un plugin (p.e. libgazebo_ros_gpu_laser)

```
$ dpkg –S libgazebo_ros_gpu_laser
$ ros-kinetic-gazebo-plugins: /opt/ros/kinetic/lib/libgazebo_ros_gpu_laser.so
```

Para listar los plugins de Gazebo que hay en este directorio

```
$ cd /opt/ros/kinetic/lib
$ ls | grep libgazebo
```

Connect to ROS

INTEGRACIÓN DE UNA CÁMARA DE PROFUNDIDAD (KINECT)

Copiar el modelo de la Kinect del directorio `~/.gazebo/models` bajo el nombre `kinect_ros`

```
$ cd ~/.gazebo/models
$ cp –r kinect kinect_ros
```

NOTA: Pueden existir varios `.sdf` del modelo. Debes incluir las mismas modificacoines en todos los archivos `.sdf` que existan del modelo.

Editar el model.sdf de la kinect_ros incluyendo el código del plugin (dentro del elemento `<sensor>`, después de `</camera>`).

… CONSULTA EL CÓDIGO PROPORCIONADO …

Connect to ROS

INTEGRACIÓN DE UNA CÁMARA DE PROFUNDIDAD (KINECT)

Arrancar Gazebo desde ROS

```
$ roslaunch myworldgazebo myworldgazebo.launch
```

Añadir a la escena el modelo de Kinect customizado (`kinect_ros`) desde la pestaña `Insert`, base de datos `~/.gazebo/models .`

Observa que se generan nuevos *topics* con información de datos del sensor

```
$ rostopic list
```

Esta información se puede imprimir por terminal

```
$ rostopic echo /camera/rgb/image_raw
```

Al ser información de profundidad e imágenes, se muestran matrices de números. Para poder obtener de una forma más intuitiva la información capturada por el sensor se puede visualizar mediante la herramienta RviZ

```
$ rosrun rviz rviz
```

Fixed Frame: /camera/link
Add – PointCloud2/Camera/DepthCloud

Connect to ROS

INTEGRACIÓN DE UN ROBOT MÓVIL

Haz una copia de alguno de los modelos de la base de datos local. Por ejemplo, del pioneer3at

```
$ cd ~/.gazebo/models
$ cp –r pioneer3at pioneer3at_ros
```

Edita los ficheros `.config` y `.sdf` para cambiar el nombre del modelo y añadir el plugin de movimiento.

```
<plugin name="differential_drive_controller" filename="libgazebo_ros_diff_drive.so">
```

… CONSULTA EL CÓDIGO PROPORCIONADO …

Añade este nuevo modelo a tu mundo de Gazebo y consulta los nuevos *topics* que se generan.

Gracias a este plugin ahora se puede controlar el modelo virtual del pioneer3at a través del *topic* `/pioneer3at_ros/cmd_vel` .

Connect to ROS

EJEMPLO COMPLETO. ROBOT MÓVIL CON CÁMARA KINECT

Crea una nuevo modelo copia del Pioneer3at creado antes (pioneer3at_ros) llamado pioneer3at_ros_Kinect. Acopla la cámara Kinect creada anteriormente (kinect_ros) a este modelo del Pioneer modificando los archivos .sdf con las siguiente líneas:

```
 <include>
      <uri>model://kinect_ros</uri>
      <pose>0.2 0 0.15 0 0 0</pose>
</include>
<joint name="kinect_joint" type="fixed">
      <child>kinect_ros::link</child>
      <parent>chassis</parent>
</joint>
```

… CONSULTA EL CÓDIGO PROPORCIONADO …

Añade el nuevo Pioneer3dx a tu mundo de Gazebo y comprueba los nuevos *topics* creados para mover el robot y para adquirir información del entorno.

Connect to ROS

CONTROL DE LA SIMULACIÓN DESDE CÓDIGO C++

Crea un archivo `.cpp` en el directorio *src* del paquete *myworldgazebo* de tu workspace.

Edita el CMakeLists.txt apropiadamente para compilar este ejecutable.

Incorpora el código correspondiente para que este nodo publique en el *topic* que te interese, en este caso mensajes de velocidad (tipo `geometry_msgs/Twist`) para controlar el robot móvil.

```
$ gedit ~/catkin_ws/src/myworldgazebo/src/navigation.cpp
```

… CONSULTA EL CÓDIGO PROPORCIONADO …

Compila y ejecuta este nodo y controlarás el robot móvil

```
$ cd ~/catkin_ws
$ catkin_make
$ rosrun myworldgazebo navigation.cpp
```

GAZEBO

David Vargas Frutos

david.vargas@upm.es – dvargasfr@gmail.com

Centre for Automation and Robotics

Centro de Automática y Robótica

Madrid, Spain

Madrid, Octubre 2017