



# [Tutorial] Modern Gazebo y integración con ros2\_control

Jonathan Cacace - EURECAT, Centre Tecnològic de Catalunya

*"innovating with businesses"*



# Presentación

- Jonathan Cacace, investigador en Eurecat (Barcelona)
  - Robòtica Cognitiva-HRI línea estratégica



**eurecat!**  
Robotics



## Unidad de Robótica @ Eurecat:

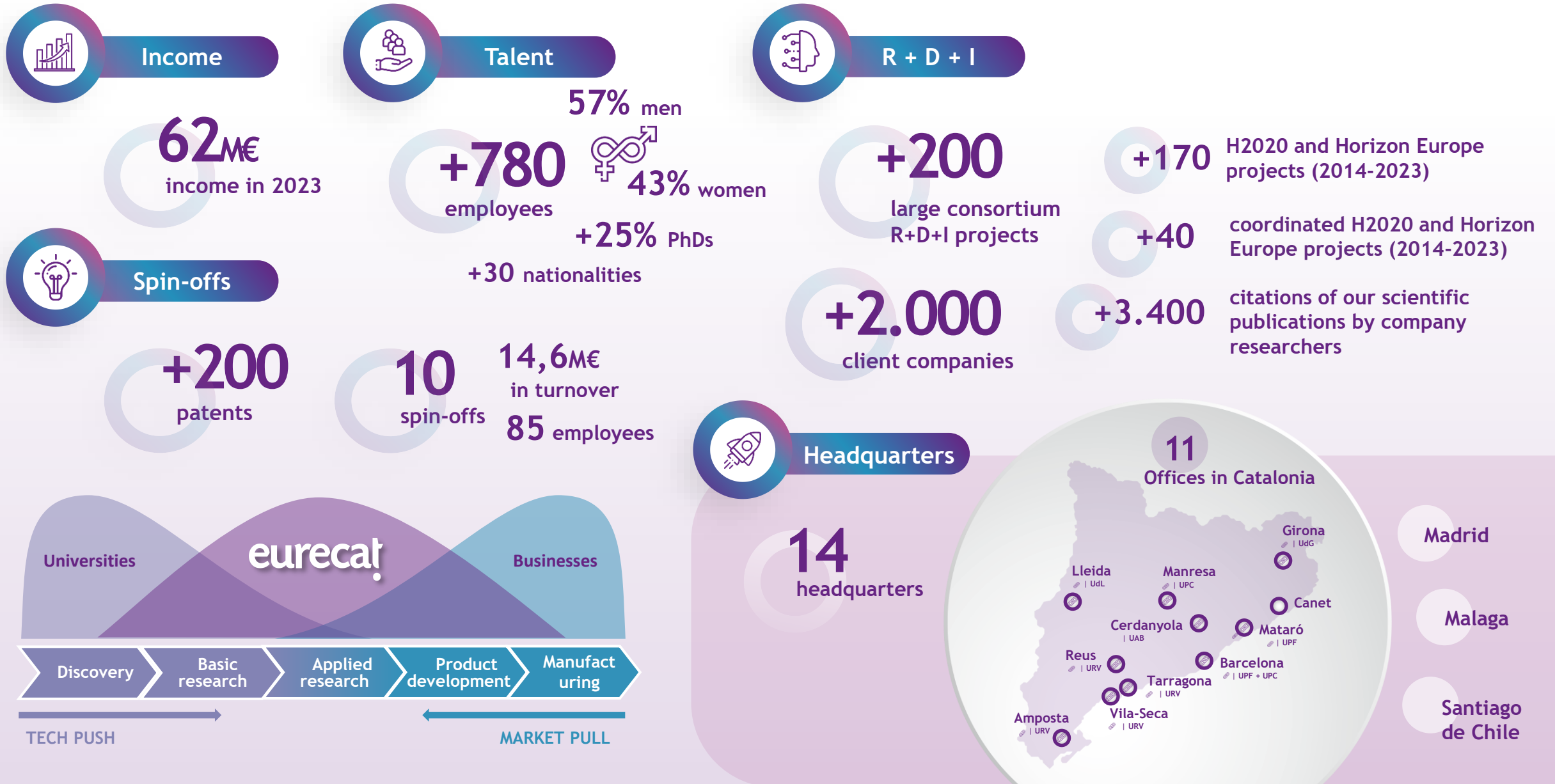
- Dirección científica: Daniel Serrano
  - Robótica Cognitiva y HRI
  - Navegación móvil
  - Manipulación y Collaborative

Since 2007   +40 researchers   6 PhDs ongoing   8 MSc Thesis yearly



(or half of it)

# Eurecat at a glance





# Robotics Research Lines

## Applied Research

R+D+I with market impact

+400

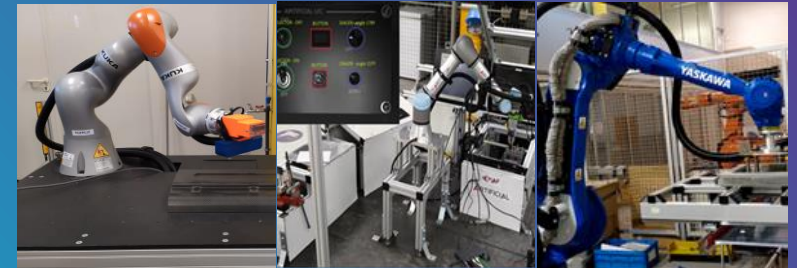
private contracts

+200

research projects

### Robotics Manipulation

- Industrial & Collaborative Robotics
- Dexterous Manipulation & Grasping
- Planning & Control - Skill Learning
- AI for Quality inspection



### Mobile Robotics

- AMR, Ground and Aerial Vehicles
- 3D and Semantic SLAM
- Indoor and Outdoor
- Human-aware Navigation



### Cognitive Robotics

- Socially Assistive Robotics
- AI Empowered Robotics
- Affective HRI/Mutual Understanding
- Reasoning and Planning



# Building a ecosystem



Collaborative R+D+I in:

- Projects with companies
- Public research projects

And ecosystem:

- DIHs, and networks
- Partnerships
- Events and competitions
- Research infrastructures



Socially  
Intelligent  
Robotics  
Consortium



Plataforma  
Tecnológica  
Española de  
Robótica

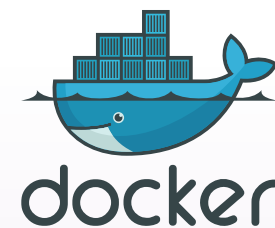


# Introducción y orden del tutorial

- Gazebo
    - Gazebo classic està al final de su vida (Enero '25) -> Modern Gazebo
      - misma filosofía
      - nuevas interfaces
      - nuevos modelos de simulación
  - Ros2\_control
    - Integración con Gazebo
    - Desarrollar nuevos controladores
  - En este tutorial
    - Modern Gazebo
      - Instalación y primeros pasos
      - Integración de nuevos robots y sensores
      - Desarrollo de complementos desde cero
    - Ros2\_control
      - Integración de controladores predeterminados con Gazebo
      - Desarrollo de nuevos controladores personalizados
-



# Requisitos previos



- ROS 2
  - Conocimientos básicos de ROS 2 (principalmente Python launch files y nodos de C++)
  - Conocimientos de URDF modeling
  - Todo el código de los ejemplos de hoy está al repo: [github.com/jocacace/ROSConES](https://github.com/jocacace/ROSConES)
    - Para instalar dependencias puede usar el Docker
    - Puede generar esta imagen con un DockerFile y iniciado con el Docker compose (todos está en le repo)
    - Si desea utilizar Jazzy (el último LTS) hay que hacer algún pequeño cambio
- ¿Qué necesitas?
  - Un laptop con Linux (si usas ROS 2 sin Dockers, con Ubuntu 22.04)
  - Esto repository: [github.com/jocacace/ROSConES](https://github.com/jocacace/ROSConES)

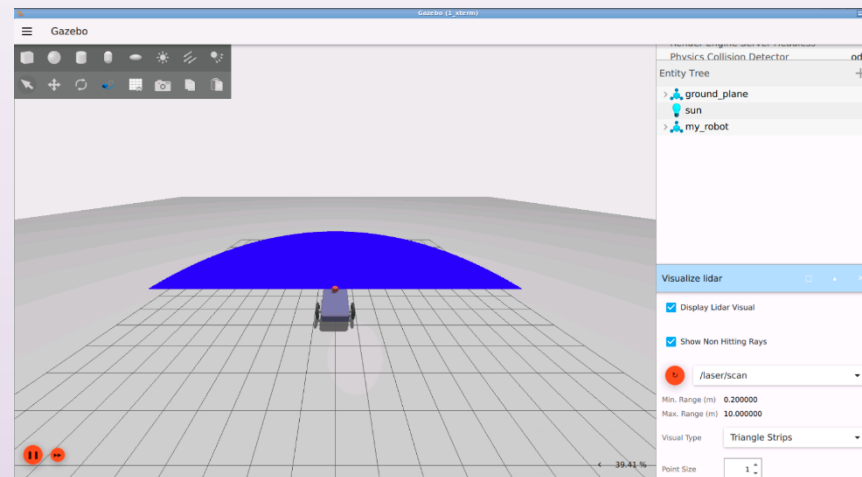
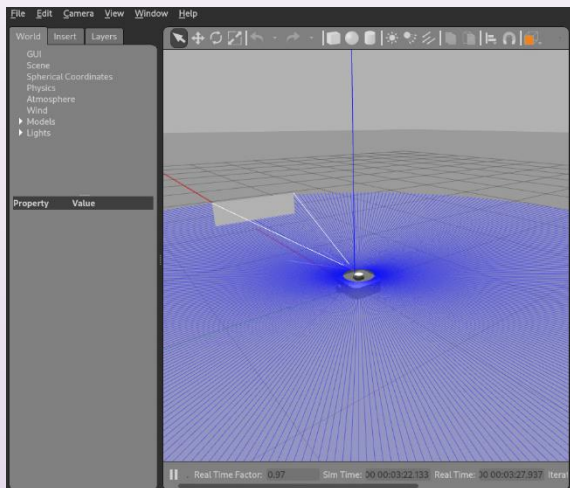
Este es un tutorial de nivel básico, el código que se encuentra aquí no es perfecto, no es optimización y no es lo mejor que puedo escribir. Pero es una guía que puedes usar en el futuro.

\* siempre puedes contactarme por correo electrónico después el tutorial ([jonathan.cacace@eurecat.org](mailto:jonathan.cacace@eurecat.org)).



# Primera parte: Modern Gazebo

- Un poquito de historia... Que es Gazebo!
- Lanzamiento inicial (2004) - Gazebo fue desarrollado inicialmente como la simulación robótica bajo el Player Project.
  - Objetivo: proporcionar una simulación 3D
- 2009: Había la integración formal de Gazebo con ROS
  - Paquete gazebo\_ros
- Ignition Gazebo (2018):
  - Para mejorar el rendimiento y el modularidad, se inició el desarrollo de Ignition Gazebo, una reescritura más moderna y modular del simulador, que se separó del ecosistema clásico.
  - Ignition (modern) gazebo está nativamente integrado con ROS 2







# Empezando

## Descargar el repositorio

Puedes descargar este repositorio en cualquier lugar en tu sistema. Se asume que lo clonarás en tu carpeta personal (home)

```
$ cd ~  
$ git clone https://github.com/jocacace/ROSConES.git
```



## Build y ejecutar la imagen

```
$ cd ROSConES/Docker  
$ docker build -t ros2roscon .  
$ docker compose up # You will lose the control of this terminal tab
```



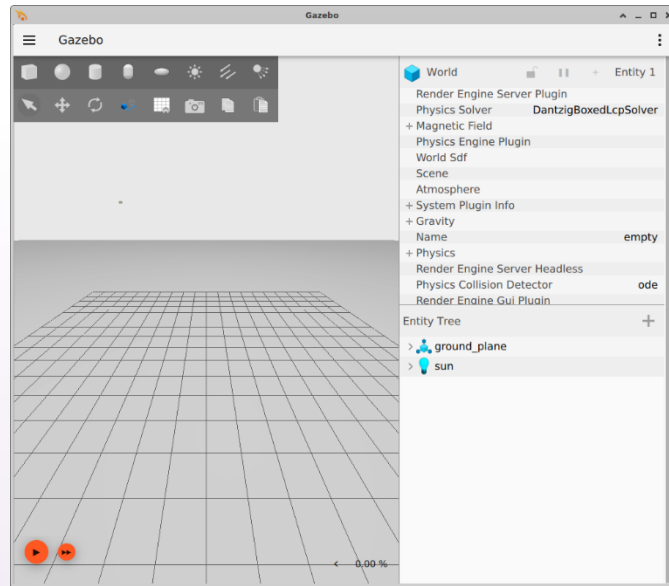
Para adjuntar un nuevo terminal a este contenedor Docker, usa el comando *docker exec*

```
$ docker exec -it ros2roscon_container bash
```





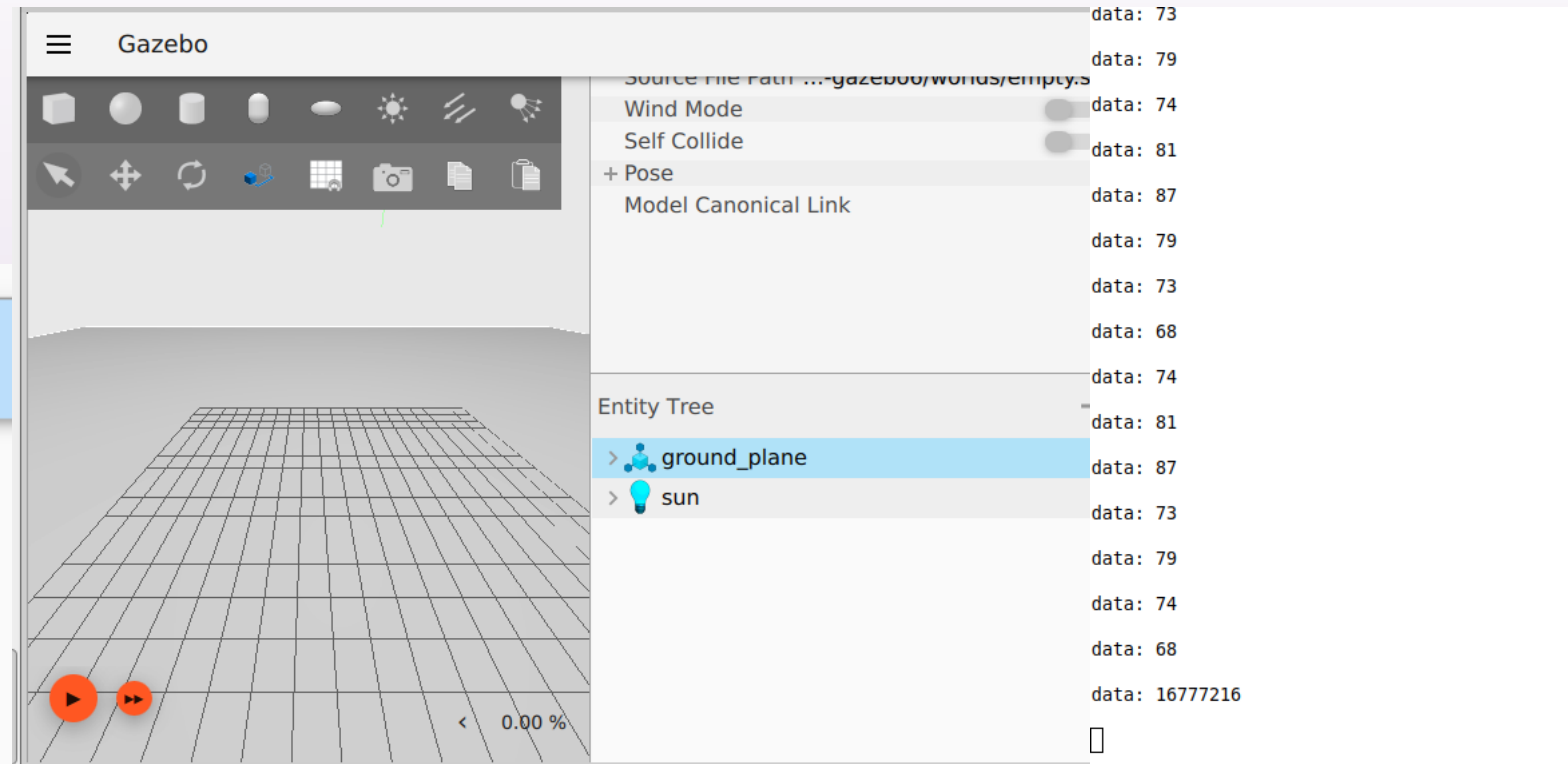
# Ejemplo 1: Key Publisher from Gazebo



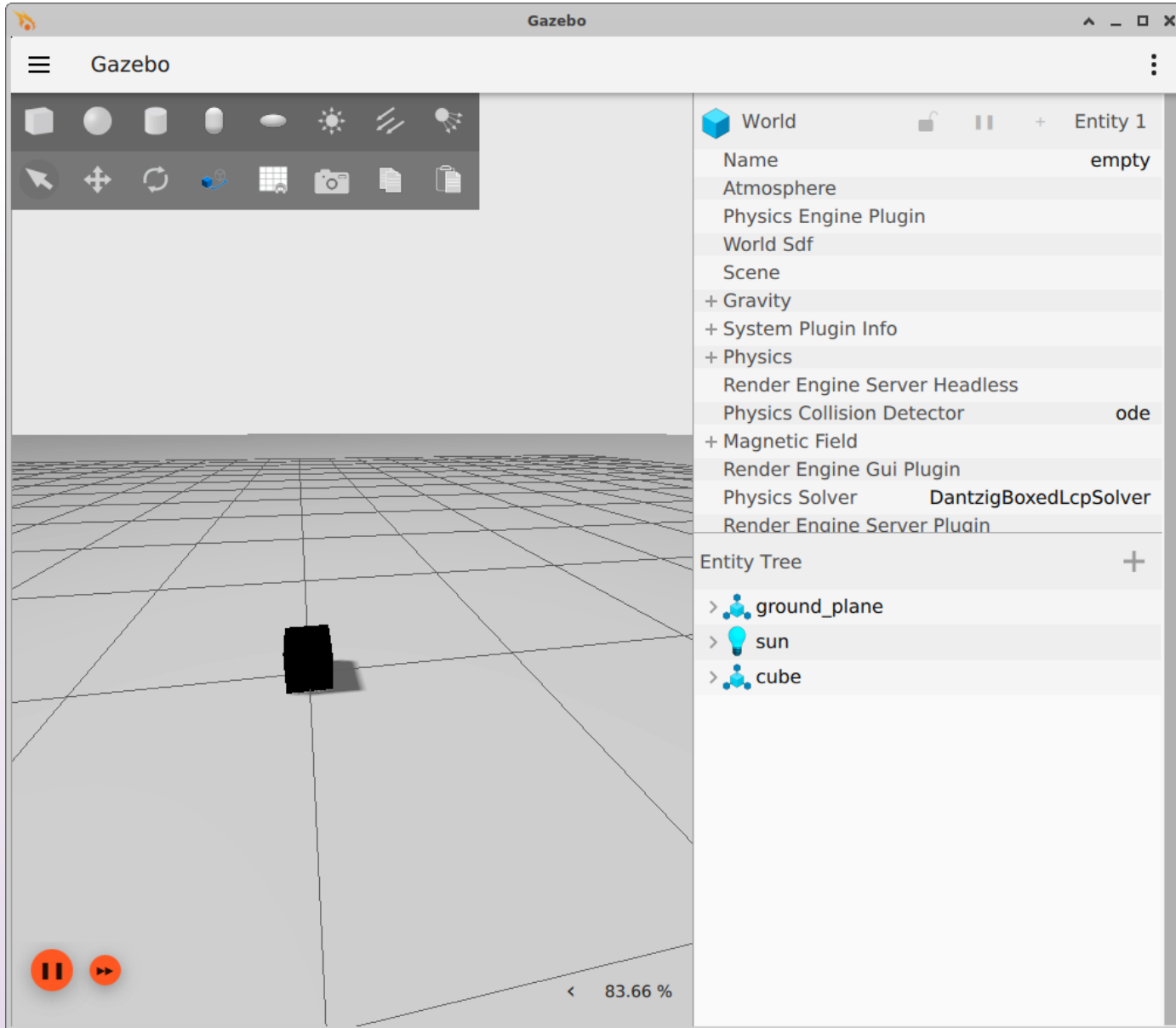
Goal: Aprende a gestionar los topics en Gazebo

## Key publisher

Keystrokes are being published to topic:  
'/keyboard/keypress'.



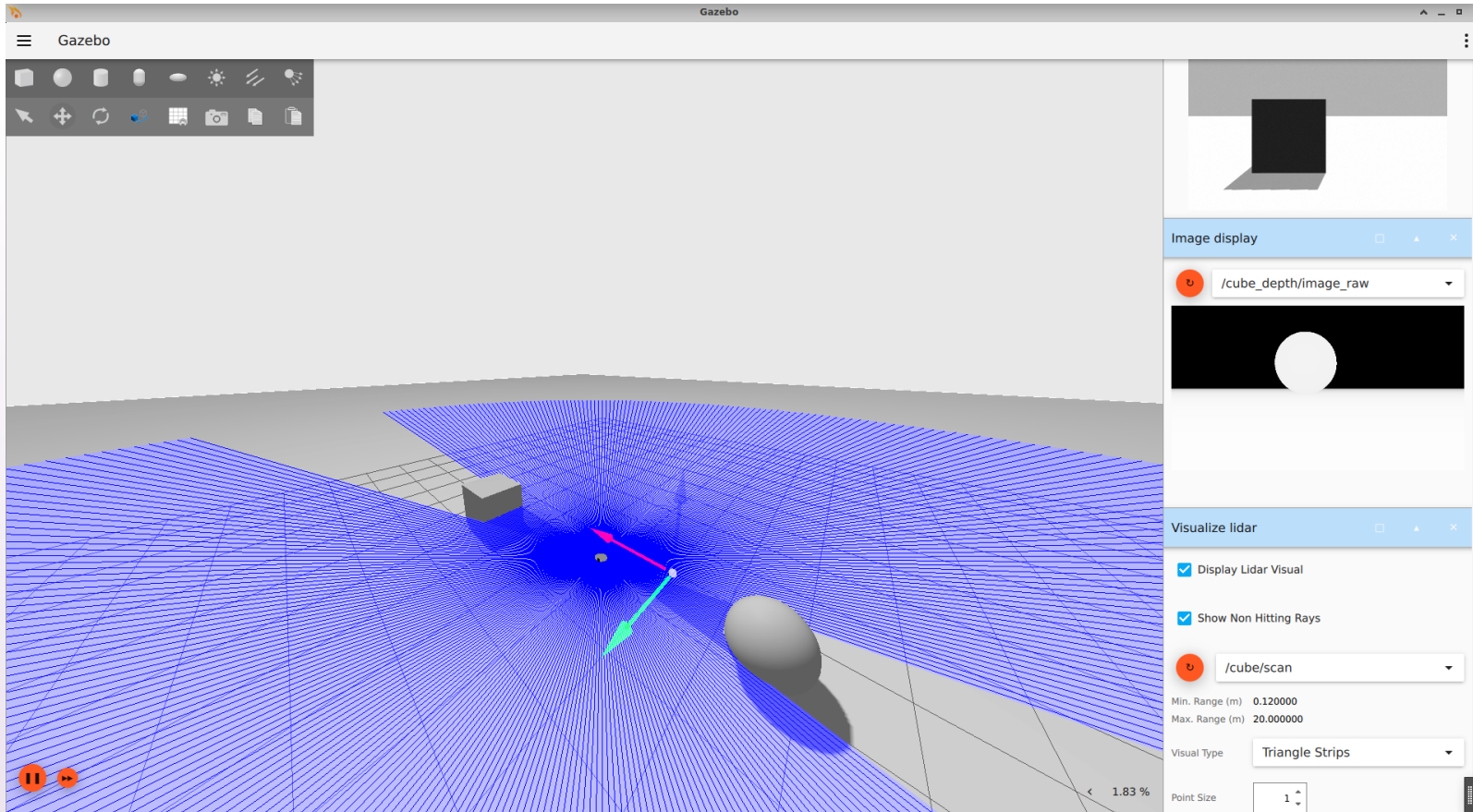
# Ejemplo 2: Anadir un objeto en Gazebo



Goal: spawn un objecto desde Gazebo

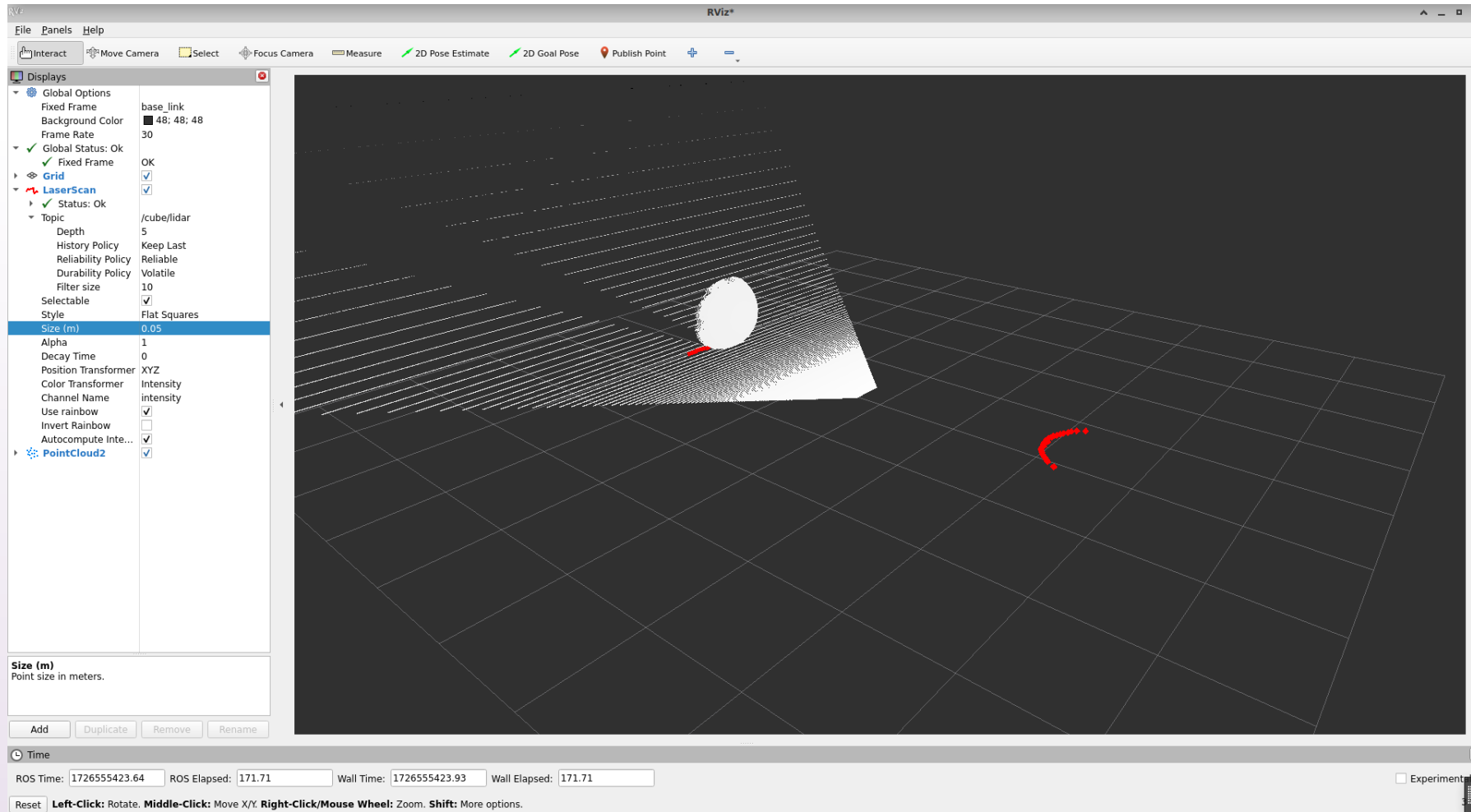


# Ejemplo 3: Anadir sensores



Goal: spawn un objecto desde Gazebo  
con diferentes sensores

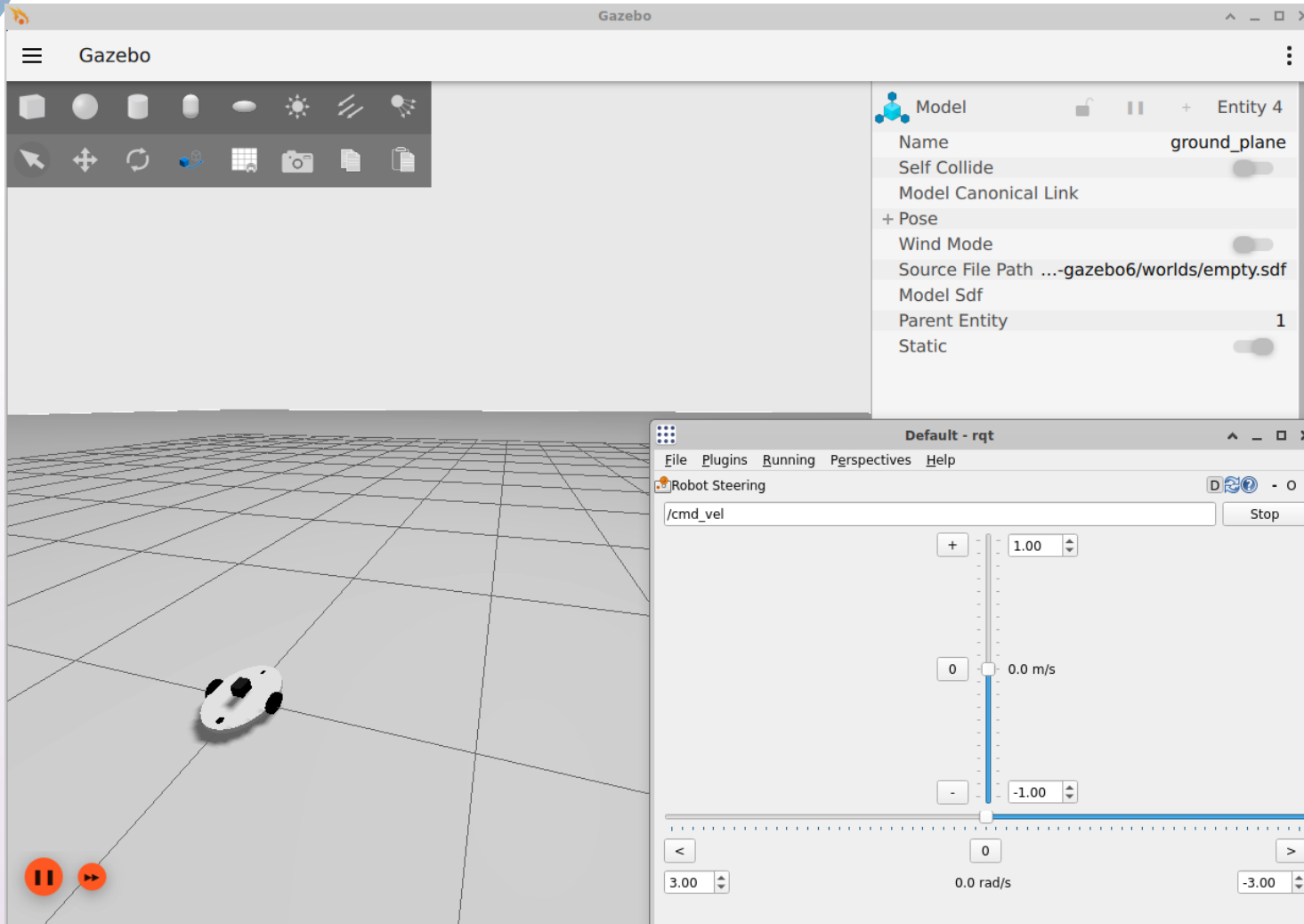
# Ejemplo 4: El ROS 2 bridge



Goal: utilizar ROS para leer los datos

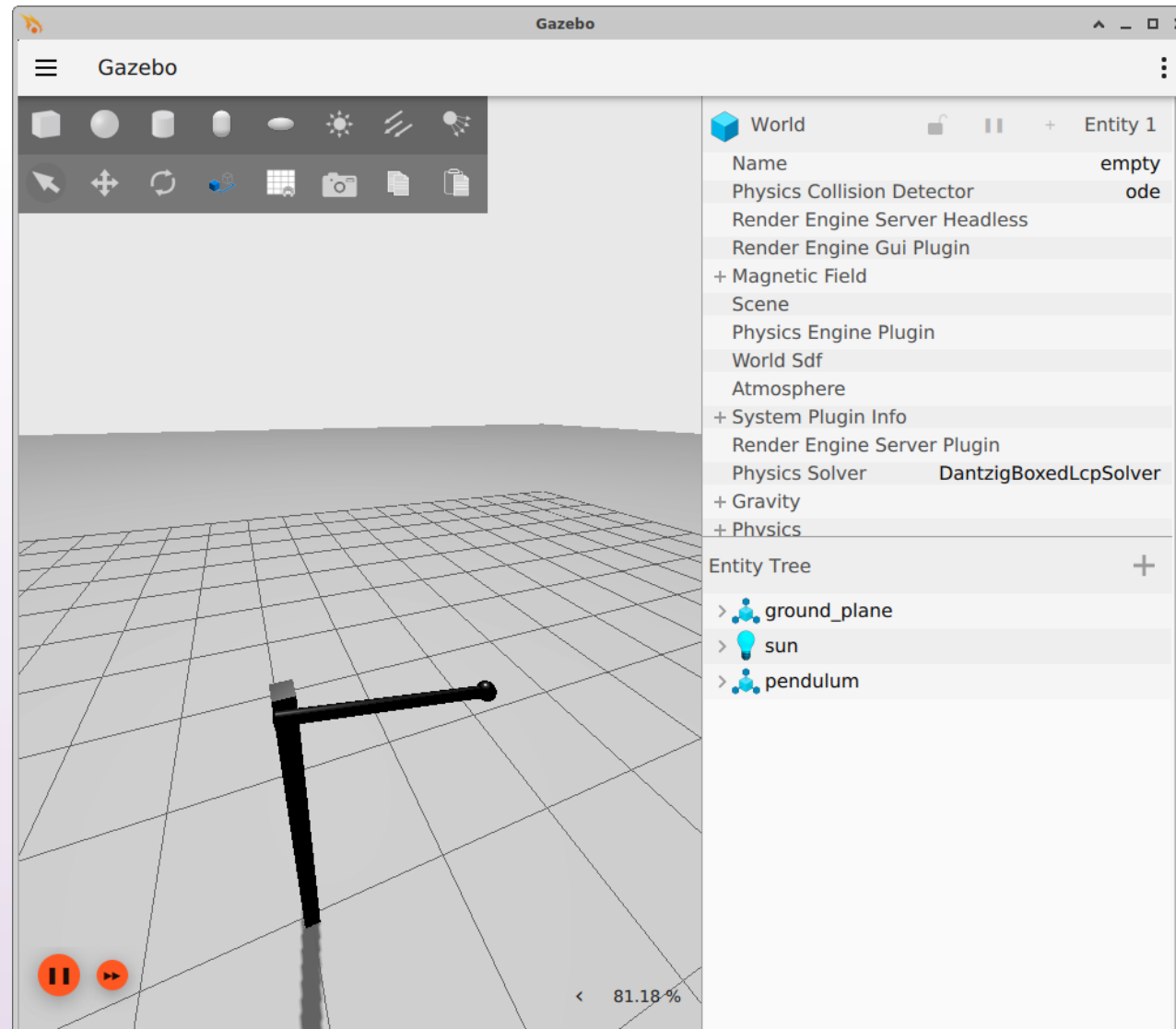


# Ejemplo 5: Differential drive robot



Goal: crear y controlar un differential drive robot

# Ros2\_control + Gazebo



# Thank You!

[Tutorial] Modern Gazebo y integración con ros2\_control

Jonathan Cacace - EURECAT, Centre Tecnològic de Catalunya

eurecat