

Dinámica de Robots.

Practica 1.

Instalación de ROS.

Ingeniería en Mecatrónica.

UPZMG.

Chagoya de la Cruz Levi Hazael.

Viorato Arámbula Alexis Israel.

Vazquez Flavio Antonio.

Hernández Morales Fco Javier.

Gomez Carrillo Christian Salvador

Introducción.

¿Qué es ROS?

Robot Operating System (ROS) es un middleware robótico, es decir, una colección de frameworks para el desarrollo de software de robots. ROS se desarrolló originariamente en 2007 bajo el nombre de switchyard por el Laboratorio de Inteligencia Artificial de Stanford para dar soporte al proyecto del Robot con Inteligencia Artificial de Stanford (STAIR2). Desde 2008, el desarrollo continuó principalmente en Willow Garage, un instituto de investigación robótico con más de veinte instituciones colaborando en un modelo de desarrollo federado.

A pesar de no ser un sistema operativo, ROS provee los servicios estándar de uno de estos tales como la abstracción del hardware, el control de dispositivos de bajo nivel, la implementación de funcionalidad de uso común, el paso de mensajes entre procesos y el mantenimiento de paquetes. Está basado en una arquitectura de grafos donde el procesamiento toma lugar en los nodos que pueden recibir, mandar y multiplexar mensajes de sensores, control, estados, planificaciones y actuadores, entre otros. La librería está orientada para un sistema UNIX (Ubuntu -Linux-) aunque también se está adaptando a otros sistemas operativos como Fedora, Mac OS X, Arch, Gentoo, OpenSUSE, Slackware, Debian o Microsoft Windows, considerados a día de hoy como 'experimentales'.

ROS tiene dos partes básicas: la parte del sistema operativo, ros, y ros-pkg. Esta última consiste en una suite de paquetes aportados por la contribución de usuarios (organizados en conjuntos llamados pilas o en inglés stacks) que implementan las funcionalidades tales como localización y mapeo simultáneo, planificación, percepción, simulación, etc.

ROS es software libre bajo términos de licencia BSD. Esta licencia permite libertad para uso comercial e investigador. Las contribuciones de los paquetes en ros-pkg están bajo una gran variedad de licencias diferentes.

Aunque está en proceso de desarrollo de aplicaciones, las áreas que ya incluye ROS son:

- Un nodo principal de coordinación.
- Publicación o suscripción de flujos de datos: imágenes, estéreo, láser, control, actuador, contacto, etc.
- Multiplexación de la información.
- Creación y destrucción de nodos.
- Los nodos están perfectamente distribuidos, permitiendo procesamiento distribuido en múltiples núcleos, multiprocesamiento, GPUs y clústeres.
- Login.
- Parámetros de servidor.
- Testeo de sistemas.

En las futuras versiones se espera que las siguientes áreas vayan apareciendo entre las aplicaciones de los procesos de ROS:

- Percepción
- Identificación de Objetos
- Segmentación y reconocimiento
- Reconocimiento facial
- Reconocimiento de gestos
- Seguimiento de objetos
- Egomoción.
- Comprensión de movimiento
- Estructura de movimientos (SFM)
- Visión estéreo: percepción de profundidad mediante el uso de dos cámaras
- Movimientos
- Robots móviles
- Control
- Planificación
- Agarre de objetos

Objetivo.

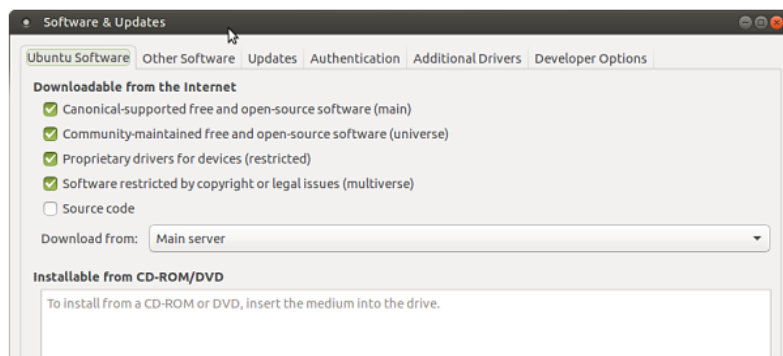
Es realizar la instalación de un marco de trabajo llamado ROS en el sistema operativo de nuestra preferencia.

Procedimiento.

Para la instalación de Ubuntu se realizó se ingresó una serie de comandos.

Step 1: Go to System -> Administration -> Software & Updates

Step 2: Check the checkboxes to repositories to allow "restricted," "universe," and "multiverse."



Step 3: Setup your sources.list

```
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" >
/etc/apt/sources.list.d/ros-latest.list'
```

Step 4: Setup your keys

```
wget http://packages.ros.org/ros.key -O - | sudo apt-key add -
```

Step 5: To be sure that your Ubuntu Mate package index is up to date, type the following command

```
sudo apt-get update
```

Step 6: Install ros-kinetic-desktop-full

```
sudo apt-get install ros-kinetic-desktop-full
```

Step 7: Initialize rosdep

```
sudo rosdep init
```

```
rosdep update
```

Step 8: Setting up the ROS environment variables

```
echo "source /opt/ros/kinetic/setup.bash" >> ~/.bashrc
```

```
source ~/.bashrc
```

Step 9: Create and initialize the catkin workspace

```
mkdir -p ~/catkin_workspace/src
```

```
cd catkin_workspace/src
```

```
catkin_init_workspace
```

```
cd ~/catkin_workspace/
```

```
catkin_make
```

Evidencias.

