

1.1. ROS Introducción

ROS es un *middleware* de código abierto (open source) que provee la funcionalidad comúnmente necesaria en el desarrollo de software para robots móviles autónomos, como paso de mensajes y manejo de paquetes. La robot Justina utiliza ROS como plataforma de desarrollo.

ROS puede describirse en dos niveles conceptuales: el sistema de archivos y el grafo de procesos.

El sistema de archivos. Se refiere al modo en que están organizados los recursos en disco:

- **Workspace:** Se refiere a las carpetas que contienen paquetes de ROS.
- **Paquete:** Es la principal unidad de organización de software en ROS. Pueden contener nodos, bibliotecas, datasets, archivos de configuración y otros.
- **Manifiesto:** Definido por el archivo `package.xml` en cada paquete. Provee meta-datos acerca de cada paquete.
- **Mensaje:** Archivos con extensión `.msg`. Definen estructuras de datos para el paso de mensajes en ROS.
- **Servicio:** Archivos con extensión `.srv`. Definen estructuras de tipo request-response. Utilizan mensajes para dicha definición.

Grafo de procesos. Es una red *peer-to-peer* de procesos. Los componentes básicos son:

- **Roscore:** Inicializa el sistema ROS: un master + rosout + un servidor de parámetros.
- **Nodos:** Es simplemente un ejecutable que usa ROS para comunicarse con otros nodos.
- **Tópicos:** Algo similar a una variable cuyo contenido puede ser compartido entre todos los nodos mediante un patrón de publicación y suscripción.
- **Servicios:** Otra forma de comunicar nodos pero con un patrón de petición y respuesta.
- **Servidor de parámetros:** Es un diccionario compartido. Todos los nodos pueden leer y escribir parámetros en tiempo de ejecución.

1.1.1. Instalación de ROS indigo para Ubuntu 14.04

Hemos creado paquetes Debian para varias plataformas de ubuntu listadas abajo. Estos paquetes son más eficientes que los creados basados en la fuente y son nuestro método preferido de instalación para Ubuntu.

Si tu necesitas instalar desde la fuente (no recomendado), por favor revisa la sección de ayuda y referencias.

1.1.2. Configura tus repositorios de Ubuntu

Configura tus repositorios de Ubuntu para permitir restringido”, “universo” y ”multiverso”. Puedes seguir ”la guía de Ubuntu”(El enlace se encuentra en ayuda y referencias) para instrucciones para hacer esto.

1.1.3. Prepara tus sources.list

Prepara tu computadora para aceptar software de packages.ros.org. ROS indigo **sólo** soporta Saucy(13.10) y Trusty(14.04) para paquetes debian.

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

1.1.4. Configura tus llaves

```
sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net --recv-key 0xB01FA116
```

Puedes intentar el siguiente comando adhiriendo **:80** si tienes el error **gpg: keyserver timed**

```
sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 0xB01FA116
```

1.1.5. Instalación

Primero asegúrate que el índice de tu paquete “Debian” este actualizado

```
sudo apt-get update
```

Si estás usando Ubuntu Trusty 14.04.2 y experimentas problemas de dependencia durante la instalación de ROS, debes instalar algunas dependencias del sistema adicionales.

⚠ Do not install these packages if you are using 14.04, it will destroy your X server.

```
sudo apt-get install xserver-xorg-dev-lts-utopic mesa-common-dev-lts-utopic libxatracker-dev-lts-utopic libopengl-mesa-dev-lts-utopic libgles2-mesa-dev-lts-utopic libgles1-mesa-dev-lts-utopic libgl1-mesa-dev-lts-utopic libgbm-dev-lts-utopic libegl1-mesa-dev-lts-utopic
```

(Do not install the above package if you are using 14.04, it will destroy your X server)
Alternativamente, intenta instalando esto para arreglar problemas de dependencia:

```
sudo apt-get install libgl1-mesa-dev-lts-utopic
```

Hay muchas diferentes bibliotecas y herramientas en ROS. proveen 4 diferentes configuraciones para que inicies. Puedes también instalar los paquetes de ROS individualmente.

Desktop-Full Install: (Recomendado): ROS, rqt, rviz, bibliotecas generales de robot, simuladores 2D/3D y percepción 2D/3D

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

Desktop install:ROS, rqt, rviz, y bibliotecas de robots en general

```
sudo apt-get install ros-indigo-desktop
```

ROS-Base: (Bare Bones) Paquetes de ROS, construcción y bibliotecas de comunicación. Sin herramientas GUI

```
sudo apt-get install ros-indigo-ros-base
```

Paquete individual: Puedes además instalar un paquete específico de ROS (reemplaza)

```
sudo apt-get install ros-indigo-PACKAGE
```

e.g.

```
sudo apt-get install ros-indigo-slam-gmapping
```

Para encontrar paquetes disponibles, utiliza:

```
apt-cache search ros-indigo
```

1.1.6. Inicializar rosdep

Antes de que puedas usar ROS, necesitaras inicializar rosdep. Rosdep le permite instalar fácilmente las dependencias del sistema para la fuente que buscas compilar y requiere correr algunos componentes del núcleo (Core) en ROS.

```
sudo rosdep init  
rosdep update
```

1.1.7. Configuración del entorno

Es conveniente si las variables del entorno de ROS son añadidas automáticamente a tu bash session cada vez que un nuevo shell es ejecutado.

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

Si tienes mas de una distribución de ROS instalada; ~/.bashrc sólo debe generarse la configuración. bash para la version que utilizas actualmente.

Si sólo buscas cambiar el entorno de tu shell actual, puedes escribir:

```
source /opt/ros/indigo/setup.bash
```

Si usas zhs en lugar de bash necesitas correr los siguientes comandos para configurar tu shell:

```
echo "source /opt/ros/indigo/setup.zsh" >> ~/.zshrc
source ~/.zshrc
```

1.1.8. rosininstall

Rosinstalla es una linea de comando frecuentemente usada en ROS que es distribuida separadamente. te permite descargar fácilmente muchos arboles de fuente para los paquetes de ROS con un comando.

```
sudo apt-get install python-roinstall
```

1.2. Instalación de PrimeSense drivers

```
sudo apt-get install freeglut3-dev pkg-config build-essential libxmu-dev libxi-dev
libusb-1.0-0-dev doxygen graphviz mono-complete
```

1.3. Instalación de OpenCV 2.4.9

```
sudo apt get update
sudo apt-get install build-essential libgtk2.0-dev libjpeg-dev libtiff4-dev
libjasper-dev libopenexr-dev cmake python-dev python-numpy python-tk
libtbb-dev libeigen3-dev yasm libfaac-dev libopencore-amrnb-dev
libopencore-amrwb-dev libtheora-dev libvorbis-dev libxvidcore-dev
libx264-dev libqt4-dev libqt4-opengl-dev sphinx-common texlive-latex-extra
```

```
libv4l-dev libdc1394-22-dev libavcodec-dev libavformat-dev libswscale-dev
default-jdk ant libvtk5-qt4-dev
cd ~
wget http://sourceforge.net/projects/opencvlibrary/files/opencv-unix/2.4.9/opencv-2.4.9.zip
unzip opencv-2.4.9.zip
cd opencv-2.4.9
mkdir build
cd build
cmake -D WITH_TBB=ON -D BUILD_NEW_PYTHON_SUPPORT=ON -D WITH_V4L=ON -D INSTALL_C_EXAMPLES=ON
      -D INSTALL_PYTHON_EXAMPLES=ON -D BUILD_EXAMPLES=ON -D WITH_QT=ON -D WITH_OPENGL=ON
      -D WITH_VTK=ON -D WITH_OPENNI=ON -D WITH_OPENCL=OFF ..
make
sudo make install
sudo echo "/usr/local/lib" >> /etc/ld.so.conf.d/opencv.conf
sudo ldconfig
```

1.4. Instalando otros paquetes de ROS

```
sudo apt-get install ros-indigo-amcl
sudo apt-get install ros-indigo-tf2-bullet
sudo apt-get install ros-indigo-fake-localization
sudo apt-get install ros-indigo-map-server
sudo apt-get install ros-indigo-sound-play
sudo apt-get install ros-indigo-pocketsphinx
```