

**INTEGRANTES DEL EQUIPO:**

JORGE ALEJANDRO SOLANO SANDOVAL

SALVADOR ALEJANDRO FARACI MACIAS

DAVID OCTAVIO PINEDO SERRANO

PEDRO IGNACIO IBARRA MERCADO

ANA YADIRA ZEPEDA ROSALES

**MARCO TEORICO**

Un robot es una máquina controlada por ordenador y programada para moverse,

manipular objetos y realizar trabajos a la vez que interacciona con su entorno. Su objetivo principal es el de sustituir al ser humano en tareas repetitivas, difíciles, desagradables e incluso peligrosas de una forma más segura, rápida y precisa.

Los robots exhiben tres elementos claves según la dentición adoptada:

- Programable, lo que signiﬁca disponer de capacidades computacionales y de manipulación de símbolos (el robot es un computador).

- Capacidad mecánica, que lo capacita para realizar acciones en su entorno y no ser un mero procesador de datos (el robot es una máquina).

- Flexibilidad, puesto que el robot puede operar según un amplio rango de programas y manipular material de formas distintas.

**ROS**

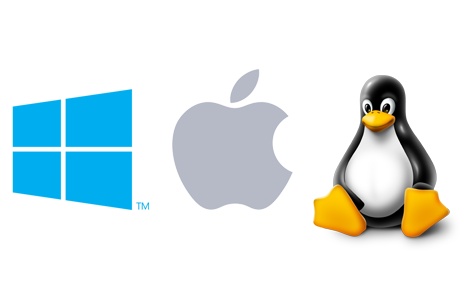
ROS (Robot Operating System) es una plataforma de desarrollo open source para

sistemas robóticos. Proporciona toda una serie de servicios y librerías que simpliﬁcan Considerablemente la creación de aplicaciones complejas para robots.

Es similar a otras plataformas de desarrollo para robots existentes en la actualidad, y su mayor virtud, es la de haber sabido aunar lo mejor de cada uno de estos sistema, juntando todo en un solo sistema capaz de comunicarse tanto con los robots más modernos, como con los ya existentes en el mercado.



Desde su creación, ROS se ha diseñado para facilitar el intercambio de software entre los aﬁcionados y los profesionales de la robótica en todo el mundo debido a su enfoque didáctico y abierto, lo que ha permitido la construcción de una gran comunidad de colaboradores a lo largo de todo el mundo.

A la hora de desarrollar, ROS permite el uso de distintos lenguajes de programación. De forma oﬁcial soportan Python, C++ y Lisp además de muchas otras como   java (todavía en fase experimental pero apoyada por google), Lua, etc.

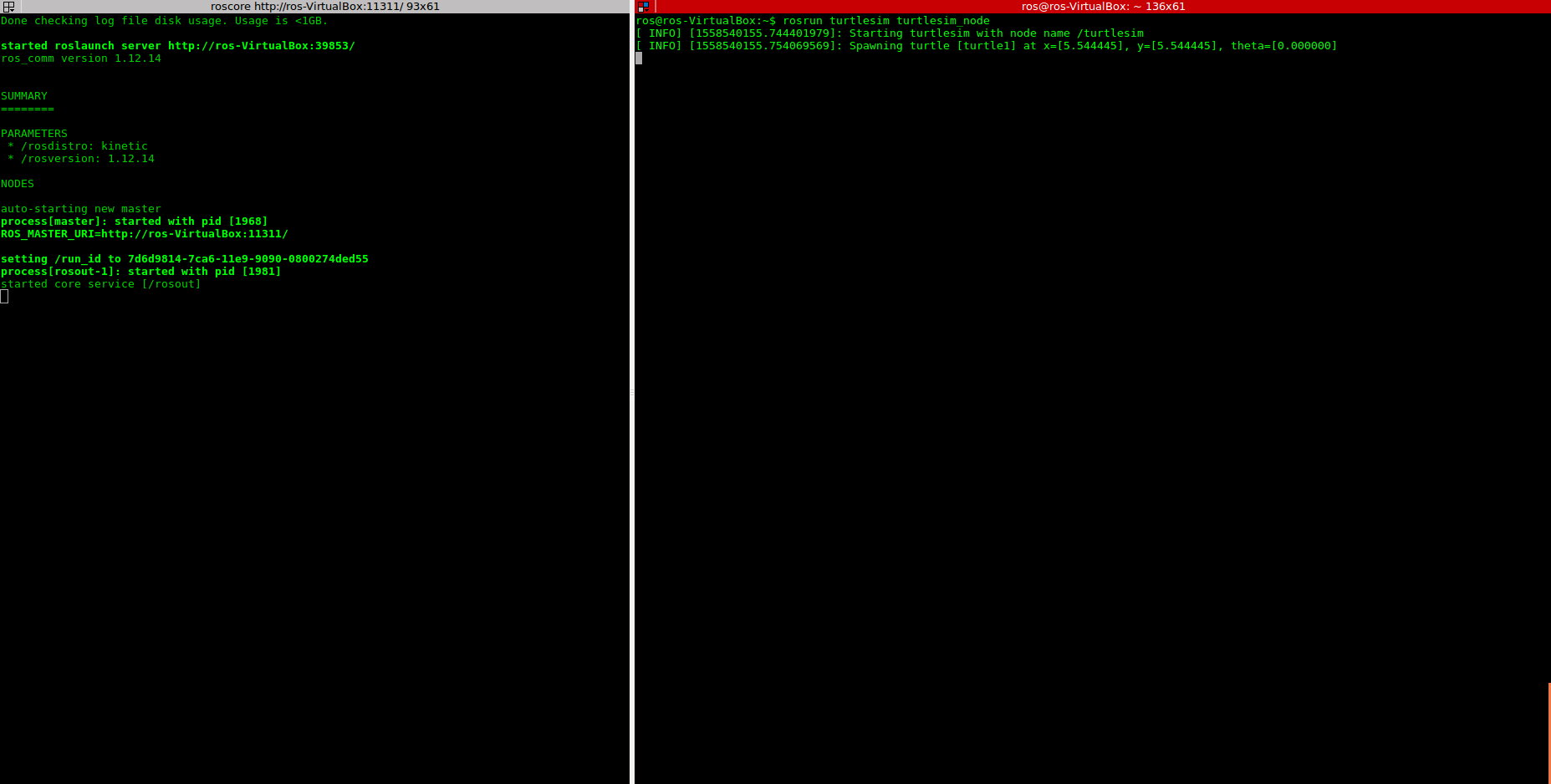
ROS puede ser ejecutado sobre maquinas tipo Unix, principalmente Ubuntu y Mac OS X aunque por parte de la comunidad puede encontrarse soporte para otras plataformas como Fedora, Gentoo, etc.

Par la instalación de ROS son simplemente los siguientes pasos:

1. Es primero que nada instalar un sistema operativo para que sea compatible con el software de ROS.
2. Se instaló el sistema operativo en máquina virtual para mayor comodidad.
3. Una vez que el sistema se tiene instalado se continuara con el software de ROS de la cual este se instalara por la terminan de Ubuntu.
   1. Para poder instalarlo se siguieron la siguiente línea de comando, comenzando por los paquetes de ROS para lo cual se agregó el siguiente comando en la terminal *sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb\_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'*
   2. Se configuraron las llaves o contraseñas con el siguiente código, *sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116*
   3. Se actualizan paquetes con: *sudo apt-get update*
   4. Se instaló la versión de escritorio de ros con el siguiente comando: *sudo apt-get install ros-kinetic-desktop*
   5. Se iniciará el ROSDEP con el siguiente código, *sudo rosdep init* y *rosdep update*
   6. La configuración del entorno mediante el código*: echo "source /opt/ros/kinetic/setup.bash" >> ~/.bashrc source ~/.bashrc*
   7. Instalación de la distribución: *source /opt/ros/kinetic/setup.bash* y *echo "source /opt/ros/kinetic/setup.zsh" >> ~/.zshrc*

*source ~/.zshrc*

* 1. Se instalaran las dependencias*: sudo apt install python-rosinstall python-rosinstall-generator python-wstool build-essential.*
  2. Por último, se abrirá el programa y se podrá trabajar con el software.

**

En la imagen de la izquierda se muestra el software ROS ya instalado y listo para su ejecución.

RESULTADOS:

A lo largo de las instalaciones diferentes obtenidas en nuestras diferentes maquinas fue una total controversia, en las diferentes maquinas ya que en una no se tuvo ningún problema al respecto de la instalación, en otra solo fue unos cambios para poder habilitar la máquina virtual en Windows, en otra máquina no se pudo colocar la máquina virtual por detalles del sistema de Windows, pero al final se logró instalar completamente funcional en dos ordenadores de los cuales se tienen para poder trabajarlos.

CONCLUSIONES:

La intalacion de ROS al principio fue muy complicada, ya que se queria instalar en el sistema debian de una raspberry pi 3, pero al ver los errores y problemas que se tenian en ella se opto por la instalacion en el sistema operativo ubuntu de linux, esto fue mas facil y mucho mas estable que en raspberry debian, al final se logro instalar sin errores y con el turtle sim funcionando.