ROS Melodic (BUILD FROM SOURCE) [Deprecated]

DEPRECATED NOTE: With the release of JetPack 5.0, the software framework has been upgraded to run ROS Noetic on Ubuntu 20. Running ROS Melodic on Ubuntu 18 is now deprecated.

ROS - melodic (ROS1) binary in Ubuntu 18.04 was compiled with opencv 3.2 , and python2 decencies. As the libraries in Ubuntu 18.04 have upgraded, it conflicts when building from deb binaries.

The solution is to build ROS from source. It is not super trivial nor consistent at times, but it works for the most part.

You might want to set your OS to default python3.

```
# Change default python variable to be python 3
sudo update-alternatives --install /usr/bin/python python /usr/bin/python2.7 2
sudo update-alternatives --install /usr/bin/python python /usr/bin/python3.6 3
```

DON'T INSTALL THE ROS APT PACKAGES, BASED ON THIS SOLUTION YOU NEED TO INSTALL ALL PACKAGES FROM SOURCE.

**HERE is the link for the tar file of the source file : Melodic Pkgs . Use the newest version and the file depending on whether this is for the Jetson or a host machine.

Common Steps:

despite the direction of the two sections below, you will need to have this initial steps for either of the two section .

```
# you will need some ROS base packages, in order to bypass the apt dependencies as a monolithic structure.
# you can use the python3 pip to install them to you system (this should go at the system level -- root)
sudo -H python3 -m pip install rosdep rospkg rosinstall_generator rosinstall wstool vcstools catkin_pkg
catkin_tools vcstool empy netifaces pycryptodomex gnupg wheel defusedxml

# ROS dependencies manager
sudo rosdep init
rosdep update

# set up a workspace ( where to build ROS from source )
mkdir -p ~/programs/ros_melodic_source_ws
cd ~/programs/ros_melodic_source_ws
```

Easy solution

- I would recommend for you to use this steps, the latter section can have unexpected issue and inconsistencies

```
# assuming you are in the workspace folder created for building ROS from source
# uncompressed tar file with the source code, this file is found above. It is copy into workspace
tar -xf ${ros_melodic_source}.tar.gz
# ensure the src directory contains all the source, then you can delete the tar file from workspace

mkdir build logs devel

sudo mkdir -p /opt/ros/melodic

# setup workspace with catkin
#   - black listed some packages that are not needed
export ROS_PYTHON_VERSION=3
sudo catkin config --init -DCMAKE_BUILD_TYPE=Release -DPYTHON_EXECUTABLE=/usr/bin/python3 -DPYTHON_INCLUDE_DIR=
```

```
/usr/include/python3.6m -DPYTHON_LIBRARY=/usr/lib/aarch64-linux-gnu/libpython3.6m.so --blacklist
rviz_plugin_tutorials librviz_tutorial --install-space /opt/ros/melodic --install

# installing dependencies

# gtk2 is needed for image_view package
sudo apt install libgtk2.0-dev
# gtk3 might be needed as well
sudo apt install libgtk-3-dev
python3 -m pip install --user -f https://extras.wxpython.org/wxPython4/extras/linux/gtk3/ubuntu-18.04 wxPython

# opencv - rosdep would pull libopencv-dev which currently point to version 4.1 but ROS melodic does not
support Opencv > 3.4.X
# Build opencv from source -- use script or manually
# --------------------
# from a folder outside your workspace, clone the following APL repo to get the build scripts
# ex: ~/programs/

git clone https://bitbucket.xrcs.jhuapl.edu/scm/slamr01/slamdev.git
cd slamdev/dependencies/

#install opencv ( identify the tag version to install in the script)
./install_opencv.sh

#--------------------------

# if building Desktop version, gazebo is needed.
sudo apt install libgazebo9-dev

# installing the remaining dependencies for python3 based on debian packages for OS
sudo apt install python3-rospkg python3-matplotlib python3-nose python3-pycryptodome python3-gnupg python3-
pyqt5.qtwebkit python3-paramiko python3-yaml python3-pil python3-sip-dev python3-coverage python3-mock python3-
pyqt5.qtopengl python3-pydot python3-pygraphviz python3-psutil python3-catkin-pkg python3-opengl python3-
netifaces python3-defusedxml python3-pyqt5.qtsvg python3-empy

# you can verify that all the python3 dependencies have been installed by the following command.
# it should return empty
export PYTHON_PKGS=$(rosdep check --from-paths src --ignore-src | grep python |  grep -v wxtool | sed -e "s
/^apt\t//g" | sed -z "s/\n/ /g")
echo $PYTHON_PKGS

# install additional packages.
sudo apt install google-mock libgtest-dev liburdfdom-dev liburdfdom-headers-dev libassimp-dev libogre-1.9-dev
libtinyxml2-dev pyqt5-dev libtinyxml-dev libcppunit-dev libpcl-dev hddtemp sbcl libbullet-dev libconsole-bridge-
dev libapr1-dev libaprutil1-dev liblog4cxx-dev libpoco-dev libcurl4-openssl-dev curl liblz4-dev libbz2-dev
libgpgme-dev libssl-dev libyaml-cpp-dev

# some python2 might be needed, probably not any more (TODO check)
# python-pyqt5.qtwebkit python-pyqt5.qtopengl python-sip-dev

# assuming that you installed opencv - user to be able to import cv2 python3 binding library
pushd ~/.local/lib/python3.6/site-packages
ln -s /usr/local/opencv_v3.4.17/lib/python3.6/site-packages/cv2 cv2
popd

# ensure that you are back in the ros workspace.
cd ~/programs/ros_melodic_source_ws
sudo catkin build
# .. hang tight .. it can take around an hour

# (optional) ros melodic linking at starter of user
echo "source /opt/ros/melodic/setup.bash" >> ~/.bashrc
```

Complete Solution for Reference mainly (Tested a while back)

- very similar to the previous step except that this time pull the packages and dependencies.

```
# assuming you are in the ROS workspace for building from source
mkdir build logs devel src

sudo mkdir -p /opt/ros/melodic


#
# pull the upstream packages and dependencies
rosinstall_generator --upstream --deps --rosdistro melodic rqt gazebo_ros_pkgs rviz image_common vision_opencv
xacro rqt_common_plugins rqt_launch rqt_plot rqt_robot_plugins rqt_image_view rqt_tf_tree rqt_srv rqt_shell
rqt_rviz rqt_robot_steering rqt_logger_level rqt_moveit rqt_nav_view rqt_publisher rqt_action rqt_py_console
rqt_reconfigure navigation_msgs pcl_msgs laser_assembler laser_geometry laser_filters realtime_tools  pluginlib
joint_state_publisher kdl_parser rosconsole ros_comm_msgs perception_pcl image_transport_plugins image_pipeline
interactive_markers orocos_kinematics_dynamics vision_opencv rqt_reconfigure navigation_msgs nodelet_core
interactive_markers rqt_msg rqt_moveit message_generation message_runtime pcl_msgs rqt_plot rqt_topic
rqt_runtime_monitor control_toolbox control_msgs rosbag_migration_rule qwt_dependency metapackages rqt_rviz
rqt_runtime_monitor roslisp urdfdom_py ros_control | tee  ./melodic_desktop_upstream.rosinstall

# Download the packages that were written into the file melodic_desktop_upstream.rosinstall
vcs import src < ./melodic_desktop_upstream.rosinstall

# ** Some packages might fail to download because the tag does not exist. You need navigate into the file
melodic_desktop_upstream.rosinstall and modify the tag name.
# The name of the tag or branch should match to name in the repository, it might be missing some prefix to the
name.
# Once you have downloaded all the source, you can check the src directory to see all the source packages.

# setup workspace with catkin
#   - black listed some packages that are not needed
export ROS_PYTHON_VERSION=3
sudo catkin config --init -DCMAKE_BUILD_TYPE=Release -DPYTHON_EXECUTABLE=/usr/bin/python3 -DPYTHON_INCLUDE_DIR=
/usr/include/python3.6m -DPYTHON_LIBRARY=/usr/lib/aarch64-linux-gnu/libpython3.6m.so --blacklist
rviz_plugin_tutorials librviz_tutorial --install-space /opt/ros/melodic --install

# installing dependencies

# gtk2 is needed for image_view package
sudo apt install libgtk2.0-dev
# gtk3 might be needed as well
sudo apt install libgtk-3-dev
python3 -m pip install --user -f https://extras.wxpython.org/wxPython4/extras/linux/gtk3/ubuntu-18.04 wxPython

# opencv - rosdep would pull libopencv-dev which currently point to version 4.1 but ROS melodic does not
support Opencv > 3.4.X
# Build opencv from source -- use script or manually
# ---------------------
# from a folder outside your workspace, clone the following APL repo to get the build scripts
# ex: ~/programs/

git clone https://bitbucket.xrcs.jhuapl.edu/scm/slamr01/slamdev.git
cd slamdev/dependencies/

#install opencv ( identify the tag version to install in the script)
./install_opencv.sh

#----------------------------

# if building Desktop version, gazebo is needed.
sudo apt install libgazebo9-dev

# you can find all
 python3 dependencies have been installed by the following command.
# it should return empty
export PYTHON_PKGS=$(rosdep check --from-paths src --ignore-src | grep python |  grep -v wxtool | sed -e "s
/^apt\t//g" | sed -z "s/\n/ /g")
echo $PYTHON_PKGS

# installing the remaining dependencies for python3 based on debian packages for OS
```

```
sudo apt install python3-rospkg python3-matplotlib python3-nose python3-pycryptodome python3-gnupg python3-
pyqt5.qtwebkit python3-paramiko python3-yaml python3-pil python3-sip-dev python3-coverage python3-mock python3-
pyqt5.qtopengl python3-pydot python3-pygraphviz python3-psutil python3-catkin-pkg python3-opengl python3-
netifaces python3-defusedxml python3-pyqt5.qtsvg python3-empy

# you can verify that all the python3 dependencies have been installed by the following command.
# it should return empty
export PYTHON_PKGS=$(rosdep check --from-paths src --ignore-src | grep python |  grep -v wxtool | sed -e "s
/^apt\t//g" | sed -z "s/\n/ /g")
echo $PYTHON_PKGS

# for reference -----
# this checks that for python2 pkgs so we can ignore them
#export IGNORE_PKGS=$(rosdep check --from-paths src --ignore-src | grep python | sed -e "s/^apt\t//g" | sed -z
"s/\n/ /g")
# sometimes this fails to ignore one to two pkgs that were passed in the skip-keys
# make sure this does not try to install ros dependencies.
#rosdep install --from-paths src --ignore-src -y --skip-keys="${IGNORE_PKGS}"
#rosdep check --from-paths src --ignore-src -y --rosdistro melodic --skip-keys="${IGNORE_PKGS}" | sed -e "s
/^apt\t//g" | sed -z "s/\n/ /g"


# install additional packages.
sudo apt install google-mock libgtest-dev liburdfdom-dev liburdfdom-headers-dev libassimp-dev libogre-1.9-dev
libtinyxml2-dev pyqt5-dev libtinyxml-dev libcppunit-dev libpcl-dev hddtemp sbcl libbullet-dev libconsole-bridge-
dev libapr1-dev libaprutil1-dev liblog4cxx-dev libpoco-dev libcurl4-openssl-dev curl liblz4-dev libbz2-dev
libgpgme-dev libssl-dev libyaml-cpp-dev

# assuming that you installed opencv - user to be able to import cv2 python3 binding library
pushd ~/.local/lib/python3.6/site-packages
ln -s /usr/local/opencv_v3.4.17/lib/python3.6/site-packages/cv2 cv2
popd

# ensure that you are back in the ros workspace.
cd ~/programs/ros_melodic_source_ws
sudo catkin build
# .. hang tight .. it can take around an hour

# (optional) ros melodic linking at starter of user
echo "source /opt/ros/melodic/setup.bash" >> ~/.bashrc
```