

# Build rosapp operating environment

---

## 1、Installation Environment

```
sudo apt-get install ros-melodic-image-transport ros-melodic-image-transport-plugins  
ros-melodic-async-web-server-cpp ros-melodic-web-video-server ros-melodic-rosbridge-  
server ros-melodic-unique-id ros-melodic-yocs-math-toolkit ros-melodic-yocs-msgs  
ros-melodic-joystick-drivers  
sudo apt install scon  
python -m pip install pymongo
```

## 2、install sdk

Copy the mongo-cxx-driver under the folder to your /home directory, and then enter the folder ,

```
cd mongo-cxx-driver  
sudo scon --prefix=/usr/local/ --full --use-system-boost --disable-warnings-as-  
errors
```

## 3、compile workspace

Create a new workspace named world\_canvas,

```
mkdir world_canvas  
cd world_canvas  
mkdir src  
cd src  
catkin_init_workspace  
cd ..
```

Then copy the contents of src under the folder to the world\_canvas/src directory, and then compile ,

```
cd world_canvas  
catkin_make
```

After the compilation is successful, add the path of the workspace to the environment variable,

```
sudo gedit ~/.bashrc
```

Add the following content to the .bashrc file,

```
source ~/world_canvas/devel/setup.bash --extend
```

Save and exit. Note that my world\_canvas is placed in the home directory, and the path here is modified according to the actual situation.

## 4、 Check whether the build is successful

Re-open a terminal and run the following command,

```
roslaunch rosbridge_server rosbridge_websocket.launch  
roslaunch world_canvas_server world_canvas_server  
roslaunch world_canvas_server map_manager.py
```