kinect

The kinect package provides nodes to acquire information from a Microsoft Kinect device.

Drivers

There are four drivers (primitives) that can be used with the Kinect. * the "'freenect"' driver (http://wiki.ros.org/freenect_launch) ** It is open source ** It does not consume a lot * the "'pure openni 1"' driver, integrated in ROS (http://wiki.ros.org/openni_camera). ** It gives you the RGB and the depth image, but not the user mask. ** It is open source ** It does not consume a lot

- the "'pure openni 2"' driver, integrated in ROS (http://wiki.ros.org/openni2_launch). ** very similar to OpenNI1
- the "'OpenNI+NITE"' driver (http://www.openni.org/files/nite/ NITE page]) ** It gives you the RGB and the depth image, and the user map (that is a nice feature!) ** It also publishes the skeleton, both as a set ROS TF transforms (http://wiki.ros.org/tf ROS TF page]) and a NiteSkeletonList custom message (https://163.117.150.59/browser/repoAD/projects/devices/kinect/unstable_ros/msg/NiteSkeletonList.msg msg file]) ** It is closed source (at least the low level NITE engine) ** It does not consume a lot

Which should I need?

It depends what you need. If you need the user map, go for NITE. Otherwise, use the freenect/openni one.

What topics do they supply?

They offer a unified interface: * "//rgb" [sensor*msgs/Image]* *"/ /depth" [sensor* msgs/Image] * "//user" sensor_msgs/Image * "//skeletons" kinect/NiteSkeletonList * "//cur*tilt*angle" [std_msgs/Float64] * "/tf" [tf/tfMessage]

Code API

Cf class `NitePrimitiveClass` and both implementations `nite_primitive_standalone.cpp` and `nite_primitive.cpp`