

1. Porting ROS packages

This page contains relevant information about how to port the existing ROS packages to a new station.

Prerequisites

1. Ubuntu or a Linux based distribution
2. ROS Indigo 14.04
3. Make sure that all the packages needed to develop Baxter programs are there such as: `baxter_simulator`, `baxter_interface`, `baxter`, `baxter_description` etc.



If the required Baxter packages are not installed go to [the following link](#) to install them.

Step-by-step guide

1. Create a ROS workspace
2. Copy or move the existing packages into your workspace (**inside the src folder**)
3. Go to the base of your package and run the `catkin_make` command
4. If all goes well you have successfully managed to port the ROS packages to a new station if not see the **What might go wrong section?**

What might go wrong ?

1. Compilation errors with packages from the NOT_REQUIRED_FOR_DEMO

NOT_REQUIRED_FOR_DEMO contains code that is written during development which might suffer frequent modifications.

Two alternatives to solve it:

1. simply **delete the folder with all its contents** and resume the make process, as the contents of this folder are available in the repository.
2. create **CATKIN_IGNORE** file inside the package folder (e.g. `touch /path/to/package/CATKIN_IGNORE`)

2. Error: xdot package not found

How to solve it: `sudo apt-get install ros-indigo-xdot`

3. Problems with the sensor driver

How to solve it: `sudo apt-get install ros-indigo-openni2*`

4. Problems with the communication

Make sure that the time step of you laptop and the robot are the same.

5. Problems with the USB channel of the sensor

A simple solution is to plug out and the plug in the sensor one more time.

6. Problems with the Py-Modbus module

Install the the module using pip or the following command `sudo pip install pymodbus`

or you can install it from the [following github address](#).

7. Errors during catkin_make

Due to the fact that the ROS packages are built on a thread based manner different issues might appear. One common error

is the fact that some packages are built while their dependencies are not yet ready, which results in errors.

A simple solution is to repeat the **catkin_make** command **several times** or to run the **catkin_make -j1** which will build the packages using a single thread, it will take longer but will do the job.

8. OpenCv or PCL not found during catkin_make

If porting the existing stack to a new computer several packages, tools might be missing, OpenCV,PCL included, which is why the errors during building are raised.

You can install them with the following command **sudo apt-get install ros-indigo-perception**

9. ROSLint package not found during catkin_make

The rviz_stream package depends on the ROSLint package. According to the official wiki the ROSLint represents "**CMake lint commands for ROS packages. The lint commands perform static checking of Python**

or C++ source code for errors and standards compliance." You can install it using **sudo apt-get install ros-indigo-roslint**