

# ICE Telescope

—

## A ROS package

Biel Artigues Aguilo

November 5, 2015

Version 0.1.1

### Abstract

*ice\_telescope* is a ROS package to operate and remote control the telescope system at the ICE building in the UAB Campus. The full system is composed of a Meade LX200GPS telescope, an SBIG ST-7 CCD camera and a Baader Planetarium dome.

## 1 Synopsis

### ROS

*roscore*

### Server

*roslaunch ice\_telescope ice\_telescope\_node*

### Client

*roslaunch ice\_telescope ice\_telescope\_node action [params]*

## 2 Description

*ice\_telescope* is composed of several nodes –*ice\_telescope\_node*– that allow the control of the telescope system. In addition to the server to control all devices *ice\_tel\_server*, each of the system components (telescope, dome, ccd) has a pair of client–server nodes following the naming convention *brand\_server* and *brand\_client*:

**Full-Server** *ice\_tel\_server*.

**Telescope** *meade\_server* and *meade\_client*.

**CCD** *sbig\_server* and *sbig\_client*.

**Dome** *baader\_server* and *baader\_client*.

**Note:** The *brand\_server* servers are there for your convenience but only the *ice\_tel\_server* is necessary to control them all.

The server node runs continuously waiting for petitions from the client node. When a client node's petition is received by the server node, the server processes the petition, sends a response back to the client and returns to the waiting mode. The client waits for the server response and finishes the execution.

### Client node

The **action** parameter issues the desired order to the server.

[**params**] will depend on the system component and the selected **action**.

**Note:** *roscore* must be running at all times for node communication and interoperation.

### 3 Servers

The servers for all the system elements are executed without additional parameters and they must be running to listen to the clients commands.

#### Full-Server

```
roslaunch ice_telescope ice_tel_server
```

#### Telescope

```
roslaunch ice_telescope meade_server
```

#### CCD

```
roslaunch ice_telescope sbig_server
```

#### Dome

```
roslaunch ice_telescope baader_server
```

### 4 Telescope client

The telescope client issues the user's desired actions to perform with the Meade LX200GPS Telescope.

```
roslaunch ice_telescope meade_client action [params]  
roslaunch ice_telescope meade_client.py action [params]
```

#### 4.1 Options

The *action* parameter is the command to be sent to the server. The *action* can be one of the following:

- |                    |   |
|--------------------|---|
| <b>goto</b>        | Point the telescope to the specified coordinates.<br><pre>roslaunch ice_telescope meade_client goto ra dec</pre> <ul style="list-style-type: none"><li>• <b>ra</b> Right ascension as a double value.</li><li>• <b>dec</b> Declination as a double value.</li></ul> |
| <b>messier</b>     | Point the telescope to the selected catalog object.   |
| <b>star</b>        | <pre>roslaunch ice_telescope meade_client messier objectNum</pre>   |
| <b>deepsky</b>     | <pre>roslaunch ice_telescope meade_client star objectNum<br/>roslaunch ice_telescope meade_client deepsky objectNum</pre> <ul style="list-style-type: none"><li>• <b>objectNum</b> The catalog number for the desired object.</li></ul>                             |
| <b>gps</b>         | Update the system's gps. <b>Note:</b> The dome must be open for the gps sync.<br><pre>roslaunch ice_telescope meade_client gps</pre>  |
| <b>getobjradec</b> | Get the coordinates of the currently selected object.<br><pre>roslaunch ice_telescope meade_client getobjradec</pre>  |
| <b>gettelradec</b> | Get the telescope's current pointing coordinates.<br><pre>roslaunch ice_telescope meade_client gettelradec</pre>  |

- getdatetime** Get the telescope's current date and time.  
`roslaunch ice_telescope meade_client getdatetime`
- setdatetime** Set the telescope's date and time to the current ones.  
`roslaunch ice_telescope meade_client setdatetime`
- getlatlon** Get the telescope's latitude and longitude.  
`roslaunch ice_telescope meade_client getlatlon`
- setlatlon** Set the telescope's latitude and longitude.  
`roslaunch ice_telescope meade_client setlatlon lat lon`
- **lat** The current latitude as a double value.
  - **lon** The current longitude as a double value.
- focus** Move the telescope's focus (in/out). WORK IN PROGRESS.

## 5 CCD client

The CCD client issues the user's desired actions to perform with the SBIG ST-7 CCD.

```
roslaunch ice_telescope sbig_client action [params]
roslaunch ice_telescope sbig_client.py action [params]
```

### 5.1 Options

The *action* parameter is the command to be sent to the server. The *action* can be one of the following:

**capture** Start an exposure and save the result to file.  
`roslaunch ice_telescope sbig_client capture filePath fileType imgCount imgType expTime readoutMode top left width height fastReadout dualReadoutChannel`

- **filePath**: The path for the saved image files.
- **fileType**: FITS or SBIG file formats.
- **imgCount**: Number of exposures to take.
- **imgType**: LF (light frame) or DF (dark frame).
- **expTime**: Number of seconds (or fraction of second) of exposure.
- **readoutMode**: Binning. Options: 1x1, 2x2, 3x3.
- **top**: Starting position in the 'Y' axis.
- **left**: Starting position in the 'X' axis.
- **width**: Image width in pixels.
- **height**: Image height in pixels.

**Note:** If all params (top, left, width and height) are zero, the full size of the CCD image is used.

- **fastReadout**: 1 for fast readout and 0 for normal readout.
- **dualReadoutChannel**: 1 for dual channel readout and 0 for single channel readout.

**settemp** Enable or disable the cooler to achieve the desired temperature for the CCD.

*roslaunch ice\_telescope sbig\_client settemp enable temperature*

- **enable**: 1 to enable and 0 to disable.
- **temperature**: double value with the desired temperature

**gettemp** Query the CCD temperature. The server returns the temperature, the power applied to the CCD as a percentage (0-1) and the cooler status (enabled/disabled).

*roslaunch ice\_telescope sbig\_client gettemp*

## 6 Dome client

The dome client issues the user's desired actions to perform with the Baader Planetarium Dome.

*roslaunch ice\_telescope baader\_client action*  
*roslaunch ice\_telescope baader\_client.py action*

### 6.1 Options

The *action* parameter is the command to be sent to the server. The *action* can be one of the following:

**open** Open the dome.

*roslaunch ice\_telescope baader\_client open*

**close** Close the dome.

*roslaunch ice\_telescope baader\_client close*

**status** Query the dome status. The possible states for the dome are: open, closed, moving and unknown.

*roslaunch ice\_telescope baader\_client status*

## 7 Files

<code>ice_tel_server</code>	C++ implementation of the server to control all devices.
<code>meade_server</code>	C++ implementation of the telescope server.
<code>meade_client</code>	C++ implementation of the telescope client.
<code>meade_client.py</code>	Python implementation for the telescope client.
<code>sbig_server</code>	C++ implementation of the CCD server.
<code>sbig_client</code>	C++ implementation of the CCD client.
<code>sbig_client.py</code>	Python implementation for the CCD client.
<code>baader_server</code>	C++ implementation of the dome server.
<code>baader_client</code>	C++ implementation of the dome client.
<code>baader_client.py</code>	Python implementation for the dome client.

## 8 Example

```
$ roscore &

$ rosrun ice_telescope ice_tel_server &

$ rosrun ice_telescope baader_client open
$ rosrun ice_telescope sbig_client settemp 1 10.0
$ rosrun ice_telescope meade_client gps
$ rosrun ice_telescope meade_client setdatetime
$ rosrun ice_telescope meade_client messier 31
$ rosrun ice_telescope sbig_client capture /img/ FITS 10 LF 30.0 1x1 0
    0 0 0 1 1
$ rosrun ice_telescope baader_client close
```

## 9 See Also

ROS, *roslaunch*, *roscd*, *rosls*, *catkin\_make*.

## 10 Requirements

**ROS Environment** *ice\_telescope* requires ROS version  $\geq 1.11.13$  ( $\geq$  Indigo distribution).

**ROS Workspace** If you want to compile or install the distributed system, you need a *catkin* workspace.

**Libraries :**

- **libusb-1.0.**
- **cfitsio.**
- **libsbigudrv.** This library can be downloaded from <http://archive.sbig.com/sbwhtmls/devswframe.htm>.

## 11 Changes

Please check the file **CHANGELOG** for the list of changes and acknowledgment to people contributing bugfixes or enhancements.

## 12 Version

Version: 0.1.1 of November 5, 2015.

## 13 License and Copyright

**Copyright** © 2015, Biel Artigues Aguilo, ICE Building, Campus UAB, Bellaterra, Catalunya  
[artigues@ice.cat](mailto:artigues@ice.cat)

The actual version of *ice\_telescope* may be found on the following link  
<https://baiels.redkaos.org/index.php/s/H4i9a87jyLQ4BMc>.

**License** This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

## 14 Author

Biel Artigues Aguiló

Email: [artigues@ice.cat](mailto:artigues@ice.cat)

Web: <http://www.ice.csic.es/>.