ICE Telescope

–

A ROS package

Biel Artigues Aguilo

September 25, 2015
Version 0.1

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
rosrun ice_telescope ice_telescope_node
Client
rosrun 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. Each of the system components (telescope, dome, ccd) has a pair of client–server nodes following the naming convention *brand_server* and *brand_client*:

Telescope *meade_server* and *meade_client*.

CCD *sbig_server* and *sbig_client*.

Dome *baader_server* and *baader_client*.

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.

Telescope
rosrun ice_telescope meade_server
CCD
rosrun ice_telescope sbig_server
Dome
rosrun ice_telescope baader_server

4 Telescope client

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

rosrun ice_telescope meade_client action [**params**]
rosrun 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:

goto Point the telescope to the specified coordinates.
*rosrun ice_telescope meade_client*Â *goto ra dec*

- **ra** Right ascension as a double value.
- **dec** Declination as a double value.

messier Point the telescope to the selected catalog object.
star *rosrun ice_telescope meade_client*Â *messier objectNum*
deepsky *rosrun ice_telescope meade_client*Â *star objectNum*
*rosrun ice_telescope meade_client*Â *deepsky objectNum*

- **objectNum** The catalog number for the desired object.

gps Update the system's gps. **Note:** The dome must be open for the gps sync.
*rosrun ice_telescope meade_client*Â *gps*

getobjradec Get the coordinates of the currently selected object.
*rosrun ice_telescope meade_client*Â *getobjradec*

gettelradec Get the telescope's current pointing coordinates.
*rosrun ice_telescope meade_client*Â *gettelradec*

getdatetime Get the telescope's current date and time.
*rosrun ice_telescope meade_client*Â *getdatetime*

setdatetime Set the telescope's date and time to the current ones.
*rosrun ice_telescope meade_client*Â *setdatetime*

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.

rosrun ice_telescope sbig_client action [**params**]
rosrun 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.

rosrun 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 an 0 for single channel readout.

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

rosrun 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).

rosrun ice_telescope sbig_client gettemp

6 Dome client

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

rosrun ice_telescope baader_client action
rosrun 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.

rosrun ice_telescope baader_client open

close Close the dome.

rosrun ice_telescope baader_client close

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

rosrun ice_telescope baader_client status

7 Files

`meade_server`	C++ implementation of the telescope server.
`meade_client`	C++ implementation of the telescope client.
`meade_client.py`	Python implementation for the telescope client.
`sbig_server`	C++ implementation of the CCD server.
`sbig_client`	C++ implementation of the CCD client.
`sbig_client.py`	Python implementation for the CCD client.
`baader_server`	C++ implementation of the dome server.
`baader_client`	C++ implementation of the dome client.
`baader_client.py`	Python implementation for the dome client.

8 Example

```
$ roscore &

$ rosrun ice_telescope baader_server &
$ rosrun ice_telescope sbig_server &
$ rosrun ice_telescope meade_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, *rosrun*, *roscd*, *rosls*, *catkin_make*.

10 Requirements

ROS Environment *ice_telescope* requires ROS version >= 1.11.13 (>= Indigo distribution).

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

11 Changes

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

12 Version

Version: 0.1 of September 25, 2015.

13 License and Copyright

The actual version of *ice_telescope* may be found on the following link
`http://www.ice.csic.es/`.

14 Author

Biel Artigues Aguilo
Email: `artigues@ice.cat`
WWW: `http://www.ice.csic.es/`.