ProjectSoft®

Astronomical institute of the Czech Republic

User manual

Astro Robotic System

date: **24. 10. 2013**
version: 1.2.0
processed: Marek Elbl
Tomáš Zeman
Tomáš Turek

www.projectsoft.cz

ProjectSoft HK a.s.
Eliščino nábřeží 375
500 03 Hradec Králové
Czech republic

CONTENT

1 INTRODUCTION 3
2 INSTALLATION 5
2.1 Operating system and RAID 5
2.2 Mirroring of data folders between PC's 7
2.3 Changing of degraded HDD 7
2.4 ARS 9
2.4.1 Set ARS server as Master or Slave 9
2.5 UPS Eaton 10
3 ARS CLIENT 10
3.1 First start 10
3.2 Application 11
3.2.1 Focusing 12
3.2.2 Camera 14
3.2.3 Guiding 17
3.2.4 Log 19
3.3 TROUBLESHOOTING 20
3.3.1 ERROR - Device/s 'CCD3, CCD3(init)' is/are unavailable 20
3.3.2 Main Camera - BLOCK EXPOSING 21
3.3.3 Remote reboot of PC's 21
4 PRIME PARAMETERS 22
4.1 MAIN CAMERA PARAMETERS 22
4.2 NAMES OF THE FILTERS 22
5 AUTOMATIC SCRIPT PARAMETERS 22
6 JSON RPC 22
6.1 get 22
6.1.1 Examples 22
6.1.2 Parameters 23
6.1.3 Return 23
6.2 set 24
6.2.1 Example 24
6.2.2 Parameters 24
6.2.3 Return 24
6.3 expose 24
6.3.1 Example 24
6.3.2 Parameters 25
6.3.3 Return 25
6.4 runscript 25
6.4.1 Example (CCD camera focusing) 25
6.4.2 Example (Autoguider camera focusing) 25
6.4.3 Example (Autoguiding) 25
6.4.4 Parameters 25
6.5 killscript 25
6.5.1 Example 25
6.5.2 Parameters 26

1 INTRODUCTION

This manual describes features of image acquisition system, automated focusing, autoguider, JSON RPC and ASCOL protocol. Structure of whole system is shown in the Fig.1 below.

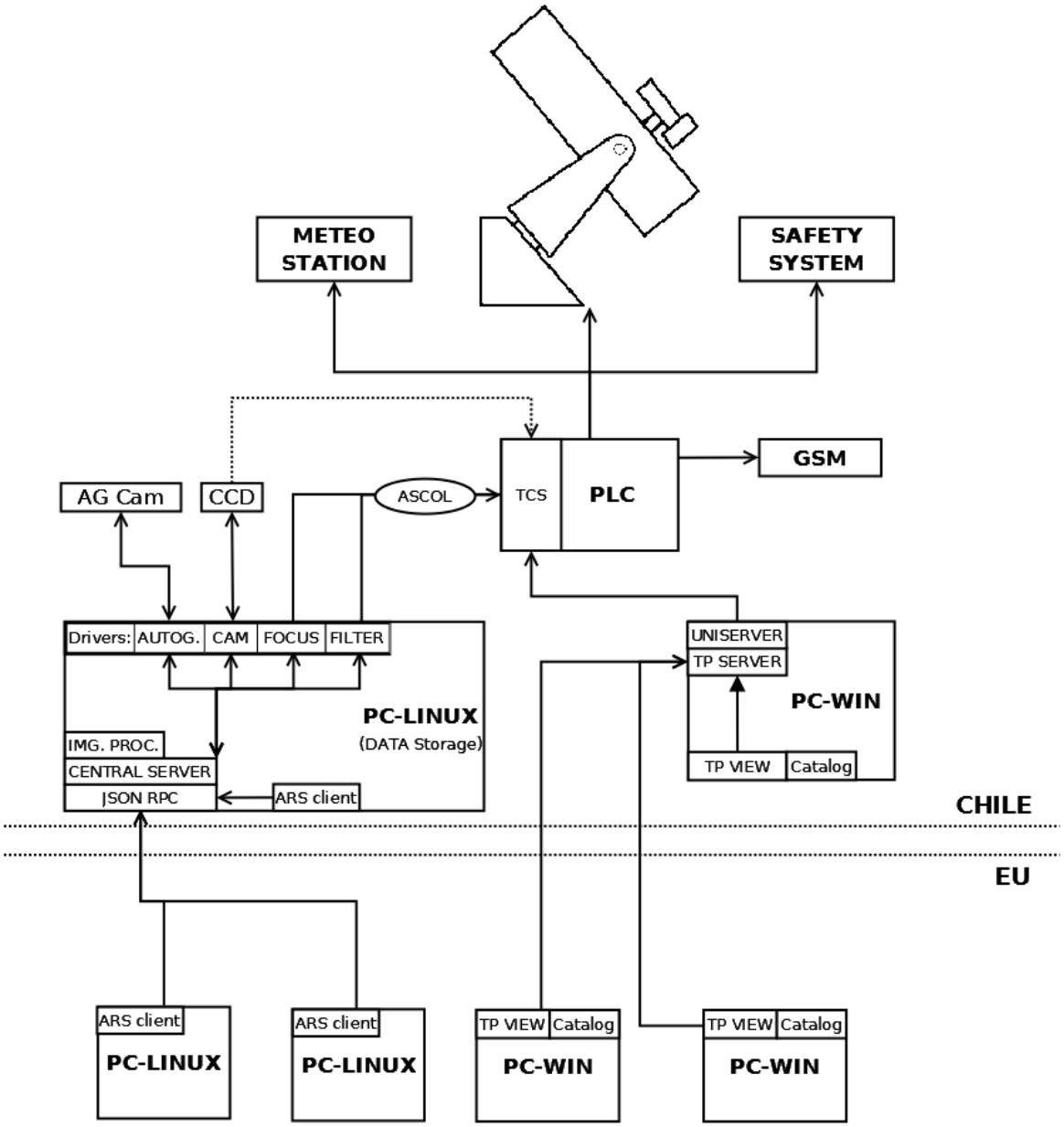

Fig.1: Structure of control and visualization system

Hardware equipment in the Telescope consists of these parts:

- PLC – “Programmable logic controller” SIEMENS
- 4 x PC – HP Z210
  - 2x with OS Linux Ubuntu 11.10 amd64
  - 2x with OS MS Windows XP SP3

Software equipment consists of these parts:

- TCS – “Telescope control system” for telescope, dome, slit, shutter, focusing, filter wheels and autoguiding movement control.
- Linux drivers for control of necessary devices:
  - Autoguiding camera (AVT)

  - Main CCD camera (CCD3)
  - Main focus (FASC)
  - Autoguiding camera focus (FGAS)
  - Telescope and autoguiding probe (TASC)
  - Filter wheel A (WASA)
  - Filter wheel B (WASB)
  - Autoguiding filter wheel (WASG)
  - JSON server (XMLRPC)

- CENTRAL SERVER – The server provides data from each available device and is built on RTS2 system. Furthermore it starts scripts for focusing and autoguiding and creates images from observing in FITS format.
- ARS CLIENT (Astro Robotic System Client) – The client allows to start reading out of CCD chip and shows last image from observing. It is able to draw an image gradually during reading out. It is possible to acquire a FITS file from last exposure and save it to path in local machine. Preview of autoguiding video signal is provided as well. Furthermore a graph of progress and quality of focusing is shown.
- UNISERVER/TP SERVER – Programs for getting data from PLC. This data are provided for visualization software
- TP VIEW – Program for visualization of the technology.

There are two communication protocols (see the protocols documentation):

- ASCOL – TCS commands via TCP/IP
- JSON RPC – JavaScript Object Notation. It allows access to Linux drivers.

2 INSTALLATION

RTS-Server runs on two redundant machines. Only one instance of RTS-Server can work at the one moment. The spare PC serves as mirror for "/data" folder where all scientific images are stored.

Servers use Ubuntu 11.10 alternate amd 64 OS.

Description of server machines:

- 2 x HP Workstation Z210 CMT Xeon E31225
- 2 x 2TB HDD as RAID 1 (mirroring) for scientific images
- 2x 500G HDD as RAID 1 (mirroring) for system

The installation of whole system is consisting of these parts:

- Install operating system "Ubuntu 11.10 alternate amd 64" + setting of RAID 1 mirroring
- Set synchronization of scientific files between two servers (Rsync)
- Install ARS (Astro Robotic System)

2.1 Operating system and RAID

Used operating system is "Ubuntu 11.10 alternate amd 64". For installation follow these steps pleas:

- PS has to include 2 x 500GB HDD (SATA0, SATA 1), 2x 2TB HDD (SATA2, SATA 3) and DVD (SATA5)
- In case of hardware RAID is set, disable it using CTLR+I immediately after boot of PC
- Boot from installation DVD
- Select English
- In the event, HDDs aren't blanked, there is necessary to erase them:
    - Choose "Rescue a broken system" after boot of installation DVD
    - Select all language settings as English … (this setting won't be used)
    - Select "Execute a shell in the installer environment"
    - HDDs should be mounted as /dev/sda, /dev/sdb, /dev/sdc and /dev/sdd
    - In the shell pleas erase "master boot record " by typping:
        - dd if=/dev/zero of=/dev/sda bs=512 count=1
    - Erase pleas "superblock" with RAID information by typing:
        - mdadm --zero-superblock /dev/sda
    - Recur this procedure in rest of HDDs
    - Reboot the PC
- Select "Install Ubuntu"
- Select all as English and US
- Set hostname: dk154-lin1 or dk154-lin2
- Select the own time zone UTC (it is at the end of list)
- Select "Partitioning method" as "Manual"
- If is it necessary please erase all md arrays
- On 500GB HDDs (sda and sdb) create a new partitions:
    - 480 GB, Primary, Beginning, Use as: physical volume for RAID, Done
    - 8 GB, Primary, Use as: physical volume for RAID, Done
- On 2TB HDDs (sdc and sdd) create a new partitions:
    - 1980 GB, Primary, Beginning, Use as: physical volume for RAID, Done
- Note - not usig the full capacity is important for the future, when only slightly smaller disk cannot be used for RAD array
- Configure software RAID
- Write the chages to the storage devices and conf. RAID: Yes

- Select “Create MD device” (it should be done three times)
- Select “RAID 1”
- Set 2 (Number of active devices …)
- Set 0 (Number of spare devices …)
- Pleas tick “Active device for RAID1 array” (it should be done in three variants):
    - /dev/sda1 and /dev/sdb1
    - /dev/sda2 and /dev/sdb2
    - /dev/sdc1 and /dev/sdd1
- Select “Finish”
- There should be new RAID1 partitions in the list.
- Set please:
    - RAID1 device #0 480 GB
        - Use as: Ext4
        - Mount point: /
        - Set format when HDD is not a new one
    - RAID1 device #1 8 GB
        - Use as: swap area
    - RAID1 device #2 1980 GB
        - Use as: Ext4
        - Mount point: Enter manually: /data
- Finish …
- Select Boot system if RAID becomes degraded: Yes
- Write changes to disk: Yes
- Select set username (dk154) and password
- Use weak password
- Encrypt home directory: No
- HTTP proxy: leave blank
- Select Install to GRUB boot loader to master boot record: Yes
- Boot to Ubuntu
- Is the system clock set to UTC: Yes
- Check network setting
    - IP: 192.168.132.53 (dk154-lin1) or IP: 192.168.132.53 (dk154-lin2)
    - Gateway: 192.168.132.1
    - DNS Server: 134.171.81.250
- After installation, enable mail sending in case of HDD will break down
    - Set appropriate SMTP connection and credentials in ~/INSTALL/raid/raidnotify.py file (line smtpserver, username, password, sender and to)
    - Copy ~/INSTALL/raid/raidnotify.py to /etc/mdadm
    - Set file executable. In terminal:
        - sudo chmod a+x /etc/mdadm/raidnotify.py
    - Edit file /etc/default/mdadm file (modify or add lines)
        - START_DAEMON=true
        - DEAMON_OPTIONS=”--syslog --program /etc/mdadm/raidnotify.py”
    - restart the PC to activation or just restart the mdadm script by command /etc/init.d/mdadm restart

- Reboot PC

2.2 Mirroring of data folders between PC's

- The mirroring copies new files from remote/master computer to the local/slave computer, but doens't deletes files which are missing on the master computer. The copy process is started at 11 AM UTC each day. Only directory /data is mirrored
- If old files need to be deleted, they must be deleted from the master PC first and then from the slave PC.
- The rsync process is started by user crontab entry, configuration is stored in file /var/spool/cron/crontabs/root
- In order to mirroring of data folder between two PC's is necessary to do steps below on each computer.
- Mirroring must be set on computer, which will be served as spare PC only (without running rts2 server)!
- Open Terminal and switch user to superuser
  - sudo bash → insert password (same as dk154 user)
- Install required packages
  - apt-get install openssh-server
  - apt-get install xinetd
- Set Public-Key Authentication (in order to passwordless connection to another PC via ssh). Open terminal and type
  - ssh-keygen –t dsa
    - Enter file in which to save the key: → Press Enter
    - Enter passphrase (empty for no passphrase): → Press Enter (Don't fill anything!)
    - Enter same passphrase again: → Press Enter
    - Your identification has been saved in /root/.ssh/id_dsa.
    - Your public key has been saved in /root/.ssh/id_dsa.pub.
    - The key fingerprint is:
    - xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx user@host
- Copy your public key to remote machine. Type into the terminal:
  - ssh-copy-id -i /root/.ssh/id_dsa.pub remoteuser@remoteipadress
    - remoteuser = dk154
- Are you sure you want to continue …: yes
- Secure local public-key in order to disable any outside access to it. Type into the terminal:
  - chmod 700 /root/.ssh/
  - chmod 600 /root/.ssh/id_dsa
- Edit crontab in order to set copying of files every minute:
  - sudo crontab -e
  - select text editor (for example 2)
- Add row below to the end of file:
    - 0 11 * * * rsync -zavrR --links --rsh="ssh -l dk154" remoteipaddr:/data /
      - remoteipaddr = should be rewritten to appropriate IP address of remote machine
- Save changes

2.3 Changing of degraded HDD

- In the event, HDDs aren't blanked, there is necessary to erase them:
  - Insert only new HDDs to the PC
  - Insert and boot from installation DVD (Ununtu 11.10 alternate amd 64)
  - Choose "Rescue a broken system" after boot of installation DVD
  - Select all language settings as English … (this setting won't be used)

    - Select “Execute a shell in the installer environment”
    - HDDs should be mounted as /dev/sda, /dev/sdb, /dev/sdc and /dev/sdd
    - In the shell pleaes erase “master boot record ” by typping:
        - dd if=/dev/zero of=/dev/sda bs=512 count=1
    - Erase pleas “superblock” with RAID information by typing:
        - mdadm --zero-superblock /dev/sda
    - Recur this procedure in rest of HDDs
- There is necessary to add partition to new HDD (we suppose sdb system disk) according to old HDD (we suppose sda system disk)
- Open Terminal and switch user to superuser
    - sudo bash → insert password (same as dk154 user)
- Show current partitions on valid HDD
    - fdisk /dev/sda
    - type “p” → press Enter (list of partitions)
    - type “q” → press Enter
- Create partitions according to “sda”
    - fdisk /dev/sdb
    - type “n” → press Enter (new partition)
    - type “p” → press Enter (primary partition)
    - → press enter (use default number)
    - type number of first sector (according to partition in “sda”)
    - type number of last sector (according to partition in “sda”)
    - repeat for every partition
    - type “t” → press Enter (change type of partition)
    - add number of partition (1 and next 2 for system disk or only 1 for data disk)
    - type “fd” → press Enter (change partition type to Linux raid)
    - repeat for every partition
    - type “w” (save changes)
- partprobe could be necessary for partition table update
- Add new partitions to RAID array:
    - mdadm –-add /dev/md0 /dev/sdb1 (double dash at the front!)
    - mdadm –-add /dev/md1 /dev/sdb2 (double dash at the front!)
- Make the new disk bootable (if the canged disk is a system one)
    - grub-install /dev/sdb

- In case the disk partition table is GPT, it is necessary to use parted instead of fdisk to create partition (example is valid for exchange of the second 2 TB data disk)
- Show current partition on valid HDD
    - parted /dev/sdc
    - type “print” → press Enter
    - type “quit” → press Enter
- Create partition according to “sdc”
    - parted /dev/sdd
    - mklabel msdos
    - mkpart p ext2 2048s 1981G (last two parameters according to partition in “sdc”)
    - set 1 raid on
    - type “quit” → press Enter

In case of the second HDD is degraded (sda) a procedure is the same, only names of sda/sdb should be changed.

In case of data HDD is degraded names of devices are sdc/sdd

Notes:
The status of the array couldbe cecked in the Disk Utility or by command cat /proc/mdstat

Resync speed of 200GB root system was 120MB/s, i.e. 30 minutes
Partitions could be also copied from one disk to other by command sfdisk -d /dev/sda | sfdisk /dev/sdb
If there is problem with mdadm --add, try increase the partition size slightly. The new member partition of the RAID must not be smaller.
If the disk controller is set to SATA or RAID mode, then the Disk Utility is capable to detect hot swap of the disk.

2.4 ARS

ARS (Astro Robotic System) is based on client-server architecture. Server side is based on RTS2 system (Remote Telescope System, 2nd Version). Whole installation is possible by using of one installation script. There is possible to install only client application of ARS or whole system with server.

- Extract installation folder of ARS to home folder (for example to /home/dk154/INSTALL)
- Installation of server and client:
  - run installation script using terminal:
    - "~/INSTALL/ars-install/ubuntu-ars-install –download-dir /home/dk154/ars-download"
  - When server will be as a master, enable auto start of rts2-server and main CCD camera driver:
    - "sudo update-rc.d rts2 default" (disable -> "sudo update-rc.d rts2 disable")
    - Enable autoload of CCD3 driver -> uncomment #ccd3 line in /etc/modules to ccd3

- Client only installation:
  - run installation script using terminal:
    - "~/INSTALL/ars-install/ubuntu-ars-client-install –download-dir home/dk154/ars-download"

2.4.1 Set ARS server as Master or Slave

In case of any problems with a server which is currently as a Master, there is possible to start the spare one.

Setting a master server to slave mode:

- Disable automatic starting of RTS2 server. Type into a terminal:
  - sudo update-rc.d rts2 disable
- Disable auto load of CCD3 driver for a main camera:
  - comment **#ccd3** line in /etc/modules to ccd3
- Edit crontab in order to set copying of files from master to slave server:
  - sudo cronttab -e
- Add or uncomment the row below at the end of file:
  - 0 11 * * * rsync -zavrR --links --rsh="ssh -l dk154" remoteipaddr:/data /
    - remoteipaddr = should be rewritten to appropriate IP address of remote machine (192.168.132.52 or 53)
    -

Setting a server in slave mode as a master server:

- Enable automatic starting of RTS2 server. Type into a terminal:
  - sudo update-rc.d rts2 enable
- Enable auto load of CCD3 driver for a main camera:
  - uncomment **ccd3** line in /etc/modules to ccd3
- Edit crontab in order to set copying of files from master to slave server:
  - sudo cronttab -e

- Comment the row below at the end of file:
  - # 0 11 * * * rsync -zavrR --links --rsh="ssh -l dk154" remoteipaddr:/data /
    - remoteipaddr = should be rewritten to appropriate IP address of remote machine (192.168.132.52 or 53)

2.5 UPS Eaton

- Install Debian installation package "~/INSTALL/ipp-linux_1.20.070-1_amd64.deb"
- Go to http://127.0.0.1:4679/ using any web browser
- Click to "Shutdown" in left column
- Click to "Edit power source" in right column
- Choose available power source

3 ARS CLIENT

This application is intended to control of the main CCD camera exposure and focusing, filter wheels setting and autoguiding of the telescope. Exposure can be set in an extended "Multi exposure" mode.

ARS client is an application with own GUI, based on python language. ARS client connects to a remote server which provides JSON RPC interface (see JSON RPC)

3.1 First start

You can start the application by typing "ars" in a terminal or using green icon on Desktop. If you start the application first time, you will have to specify an IP address of server, login and password (see Fig. 2) in the other case just choose appropriate server.

Fig. 2: Login to remote server

In case of first start of the application you will have to select appropriate department, which belongs to your work place (see Fig. 3). You can select one of these options:

- Brno
- Copenhagen
- La_Silla
- Ondrejov_NEOS
- Ondrejov_Gaia
- Praha
- ProjectSoft

This setting is possible to change in menu bar "Edit -> Department", but only with administrator account.

Fig. 3: Department selection

The department setting is bound with user management in main visualization. Control of the system is enabled only when a user is logged in main visualization. Furthermore the user must belong to appropriate department.

3.2 Application

You can see a welcome screen after correct login (see Fig. 4).

Fig. 4: welcome screen of ARS Client

Status information is placed in the upper part. The application is divided into 5 tabs regarding to main tasks:

- Welcome screen
- Cameras focusing
- Camera control
- Autoguiding
- Events log

3.2.1 Focusing

Focusing tab is divided into 4 parts:

- *Current state:* Shows general information about auto-focusing script, focuser position and date of last focusing. There is possible to change setting of an automated focusing script by button “Setting”
- *Graph:* Shows median of Full width at half maximum value (FWHM) of stars connected with focuser position. FWHM values are acquired from focusing image.
- *Commands:* This panel allows set position of focuser manually.
- Automated focusing can be started by “Focus” button. Interruption of automated focusing is possible by button “Stop”

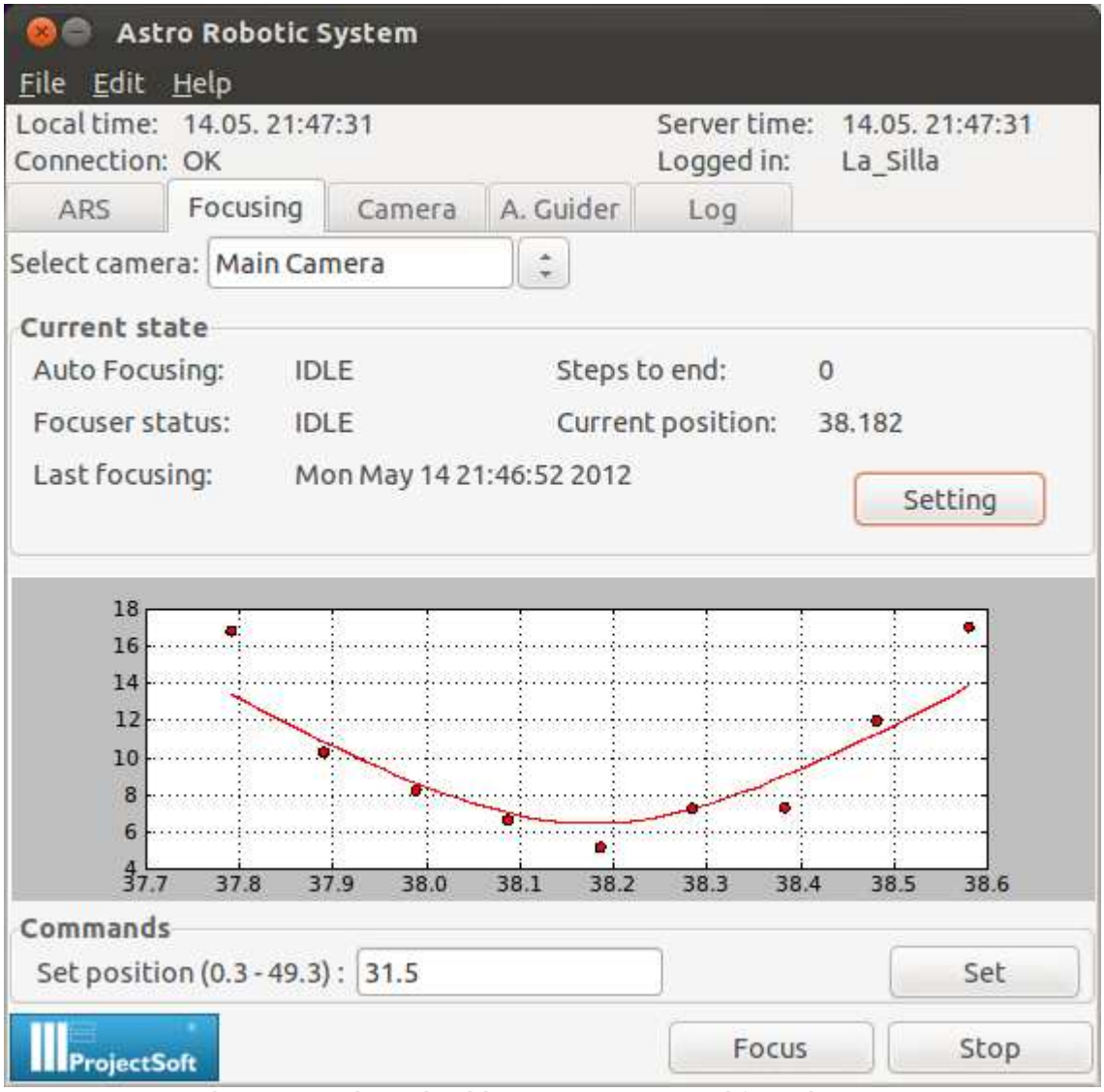

Fig. 5: Focusing tab with current automated focusing.

Automated focusing of Main Camera: Automated focusing script can be started by button “Focus” at the bottom of the focusing tab. The script consists of these steps:

- Disable clearing of CCD chip after reading out.
- Do expose, do slight move by telescope in RA axis and change of a focus position gradually according to parameters (see below)
- After last exposure do readout of whole CCD chip
- Find starts in image and compute their FWHM value
- Find focusing sequence of stars in line

- Get median of FWHM value of stars for each position of the focus
- Do fit of a hyperbolic curve to FWHM/position data
- Find minimum of the hyperbolic curve
- Move the focus to minimal position

Automated focusing of Autoguiding camera: Automated focusing script can be started by button “Focus” at the bottom of the focusing tab. The script consists of these steps:

- Do expose and change of a focus position gradually according to parameters (see below)
- Process all images from path “/data/autoguiding/focusing”. Find starts in images and compute their FWHM value
- Find each one star on all images
- Get median of FWHM value of stars for each position of the focus
- Do fit of a hyperbolic curve to FWHM/position data
- Find minimum of the hyperbolic curve
- Move the focus to minimal position

Assuming that minimal position is out of range of the focus positions range, the focusing script does estimate of minimal position.

There are two modes of automated focusing:

- *Full mode*: In this case the focusing script will start with parameters for “Full mode” (it should be wider range of focus positions) and after that the script will start again with parameters for “Small mode”. The “Full mode” starts from position which is specified parameters.
- *Small mode*: The focusing script will start only once with “Small mode” parameters. It means that a start position for focusing corresponds to current position.

Parameters: Setting window will appear after pressing of “Setting” button in focusing tab (see Fig. 6)

Focusing setting window is divided into 3 parts:

- *Small*: Setting for “Small mode” of an automated focusing script
- *Full*: Setting for “Full mode” of the automated focusing script
- Selecting of mode for the automated focusing script, setting exposure time for each expose, a button “Set” for sending of the parameters to a server.

Fig. 6: Window with parameters of automated focusing

3.2.2 Camera

Camera tab is divided into 3 parts:

- ***Current state:*** Shows general information about CCD camera setting:
  - ***X, Y begin:*** Coordinates of left bottom corner of window on a CCD chip for reading out
  - ***X, Y size:*** Size of window on a CCD chip for reading out
  - ***Binning*****:** Binning from "1 x 1" to "4 x 4"
  - ***Exposure time***
  - ***Exposures to end:*** *Number of exposures in a queue*
  - ***Multi-exposure:*** Disable or enable multi exposure
  - ***Clear before exposure:*** If it is enabled, chip will be cleared before exposure. (it is used for automatic focusing)
  - ***Filter A, B state*****:** IDLE state allows exposure. If a filter is in MOVE state, exposure will be blocked
  - ***Filter A, B*****:** Name of current filter
  - ***Channel 1, 2*****:** It determines which channel/amplifier of CCD camera will be used
  - ***Image type*****:** Determines one of these types of exposure:
    - *LIGHT – shutter open, exposure time > 0*
    - *DARK – shutter closed, exposure time > 0*
    - *BIAS – shutter closed, exposure time ~ 0*
    - *FLAT – shutter open, exposure time > 0*
  - ***Shutter:***
    - *LIGHT – shutter is opened during exposure*
    - *DARK – shutter remains closed during exposure*
  - ***Temperature:*** Temperature of CCD.
  - ***Object:*** Object name. Final image name is created according to this name.
- ***Image:*** *Date of last exposure and name of fits file which was saved in server and optionally on local machine.*
- Exposure can be started using ***"Expose"*** button. Exposure, image preview and download interruption is possible using "Stop" button

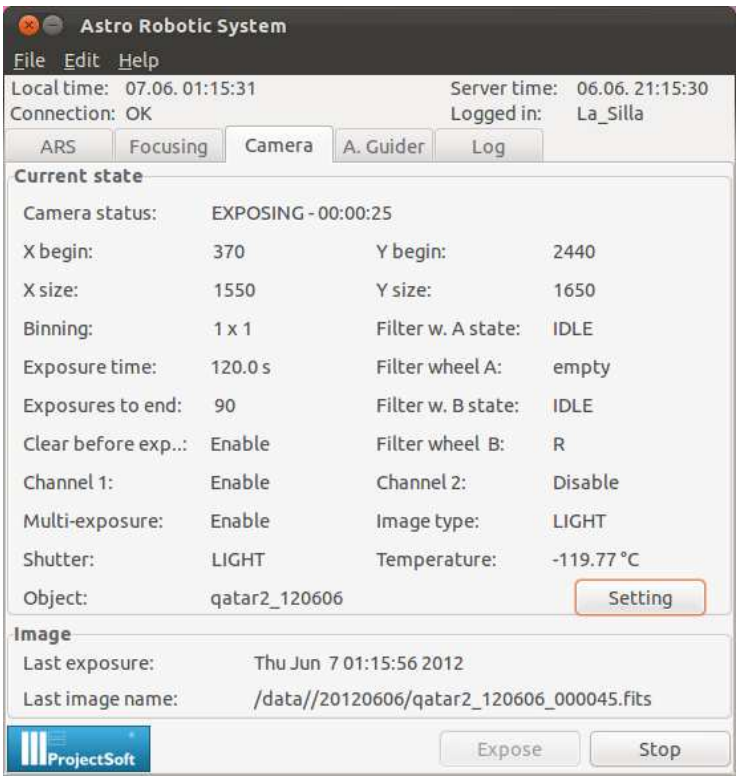

Fig. 7: Camera tab with current camera values

Parameters: Setting window will appear after pressing of "Setting" button (see Fig. 8)

Camera setting window is divided into 2 parts:

- ***Binning***: Select binning from "1 x 1" to "4 x 4"
  - *Set:* Send setting of binning to the server
- ***Camera setting****:* It allows to set parameters of exposure, reading out and filter wheels
  - ***X, Y begin:*** Coordinates of left bottom corner of window on a CCD chip for reading out
  - ***X, Y size:*** Size of window on a CCD chip for reading out
  - ***Filter A, B*****:** Number of filter
  - ***Exposure time:*** Time of exposure in seconds
  - ***Setting:*** It shows window for setting of multi-exposure.
  - ***Clear before exp (for Technologist only).:*** *It disables or enables clearing of chip before an exposure.*
  - ***Channel 1, 2 (for Technologist only)****: It specifies which data channels/amplifier will be used. Multi-cannel FITS file will be created in case of using both of channels.*
  - ***Object name****:* This name is used for creating of FITS files. Names of FITS files are created in this way: "objectname_%6d". Number of a FITS file is incremented with every exposure.
  - ***Image type****:* Determines one of these types of exposure:
    - *LIGHT – shutter open, exposure time > 0*
    - *DARK – shutter closed, exposure time > 0*
    - *BIAS – shutter closed, exposure time ~ 0*
    - *FLAT – shutter open, exposure time > 0*
  - **Local path**: It shows path on client side where will be created a folder with *FITS files.* Name of folders are consist of a current date and are created in *this way: "yyyymmdd"*
  - ***Show preview during reading (for Technologist only):*** *Enable or disable this feature*

- ***Show resulting fits file (for Technologist only):*** *This setting will cause, that a resulting FITS file will be downloaded from server to client PC (after preview is finished) and will be shown in DS9*
- ***Browse (for Technologist only):*** It allows setting of local path where FITS files will be saved.
- ***Create local copy of FITS file (for Technologist only):*** *D*isables or enables saving of last downloaded FITS file to the specified local path.
- ***Use multithread FTP (for Technologist only):*** Enable or disable FTP protocol for image download. It uses 20 independent threads and it could speed up download speed in case of poor connection quality.
- ***Set****:* Send setting to the server

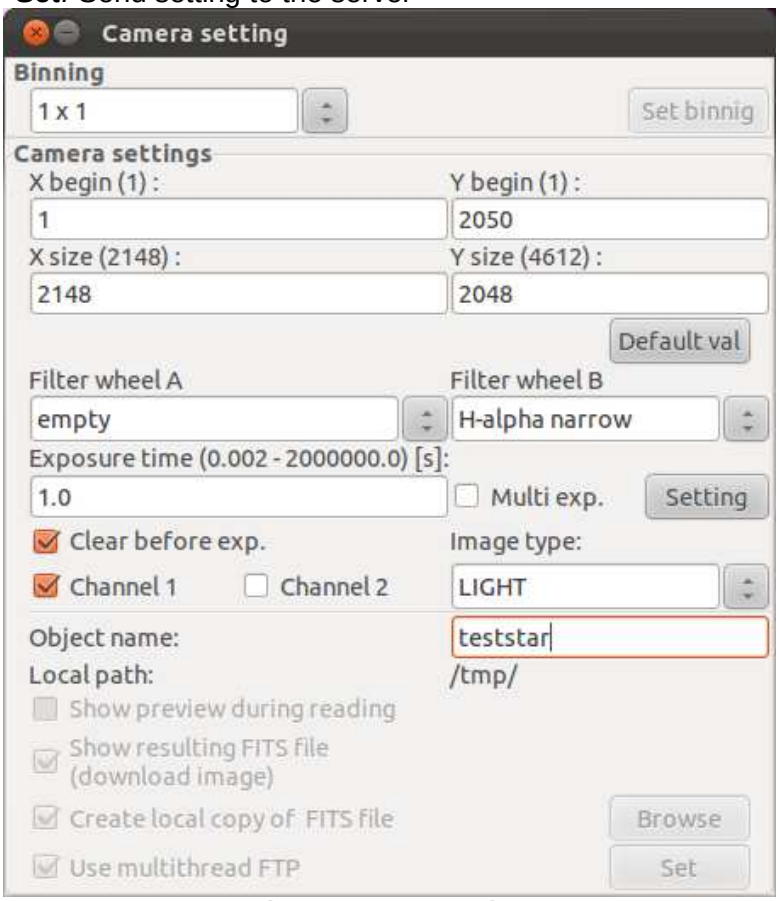

Fig. 8: Camera setting

Multi exposure: ARS-Client allows set a sequence of exposures with different filters. It is called multi-exposure. Setting of the multi-exposure is possible in Multi-exposure setting window (see Fig. 9). There is possible to specify six of different settings. Settings are processed gradually. At last there is possible to set count of repeating all of these settings. Each setting specifies:

- Whether to use this row
- Filter A, B
- Exposure time
- Count of exposures

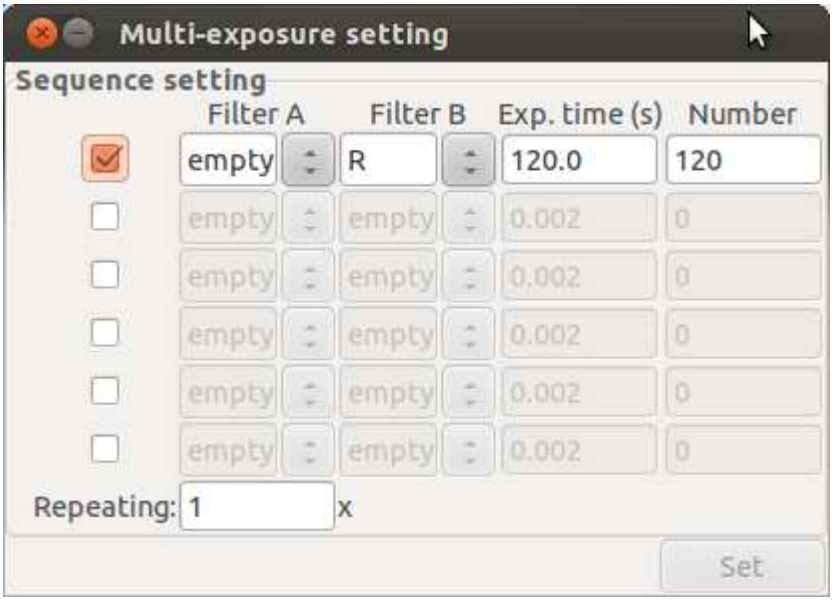

Fig. 9: Multi-exposure setting

3.2.3 Guiding

Guider tab includes data important for successful autoguiding.

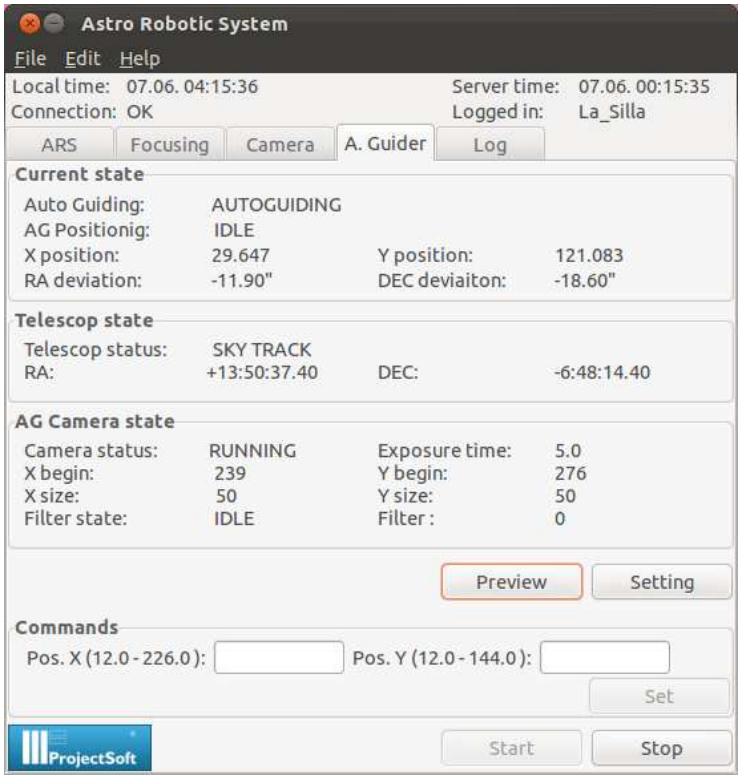

Fig. 10: Preview of autoguiding

First panel “Current state” shows state of autoguiding script. There can be only “PREVIEW” mode or “AUTOGUIDING” mode. AG Positioning includes information whether is autoguiding probe in “POSITIONING” state or “IDLE” state. When AG Positioning is in “NOT READY” state, there is necessary to initialize it in visualization of telescope. Furthermore the panel includes position information about the guide probe and RA, DEC deviation in arcsec, which was integrated during autoguiding.

Panel “Telescope state” includes label with telescope state and current RA, DEC coordinates.

“AG Camera state” includes information whether camera is “RUNNING” or “IDLE”. Size of reading out area, exposure time, filter state and current filter.

Button “Preview” opens window, which allows monitoring progress of autoguiding or image from camera only in case of Preview mode see Fig. 12.

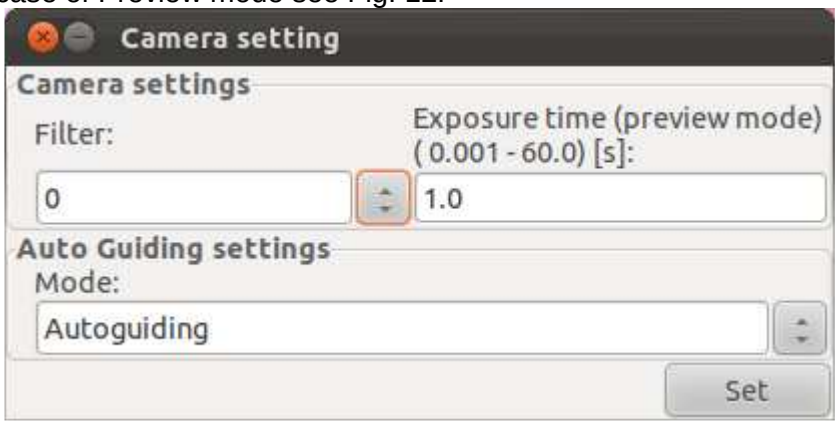

Fig. 11: A. Guider setting window

Button “Setting” opens window, which allows setting filter for autoguiding camera, exposure time (but only for preview mode) and mode. There is possible to set two modes:

- AUTOGUIDING. In this mode, autoguiding will start, after press of “Start” button at the bottom of the A. Guider tab.
- PREVIEW. There will be only image data from camera shown in preview window.

Panel “Commands” allows setting position of autoguiding probe manually. For start of autoguiding probe positioning, please insert new values (according to limits in brackets) and press “Set” button.
Button “Start” starts mode, which was preset in setting window.

Button “Stop” stops current activity.

Autoguiding consists of next steps:

- Finding star in catalogue and estimate of appropriate position of AG probe
- Positioning of AG probe
- Searching of star on full frame image
- Lock 50x50 pixels window on the brightest star
- Start autoguiding on the star.

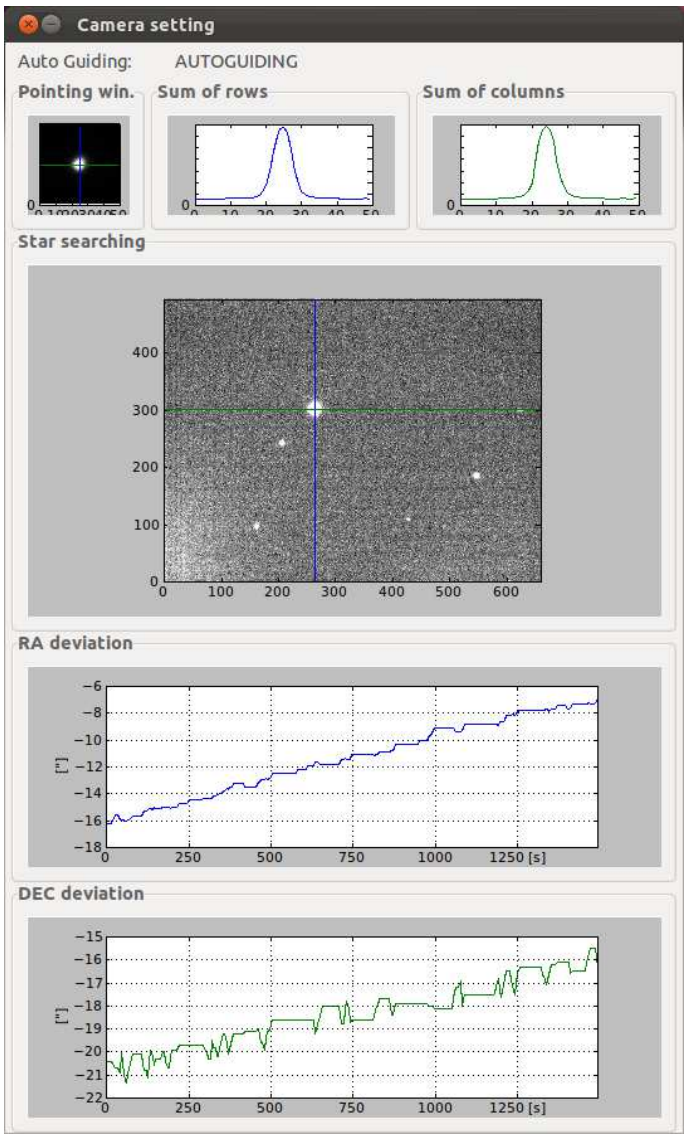

Fig. 12: Preview window of autoguiding

3.2.4 Log

Logging starts immediately after ARS-Client gets run. Log tab includes logging information from start of ARS-Client. All logging information is stored to files on local HDD to path: "~/ars-client/logs". Logging system distinguishes two levels of logging information:

- *ERROR:* This type of message will appear when some device, driver, server or any other module fails. There are examples of ERROR messages:
  - *Permission denied:* When a path, to folder where FITS files will be saved, is not owned by current user.
  - *Device 'XXXX' is unavailable:* When a driver of a device brakes down. 'XXXX' means name of the driver in RTS-Server. In case of this message appears you can wait until the driver will start automatically again (approximately 1 minute) or you can restart the driver manually. Another option is to restart RTS-Server (by typing "sudo service rts2 restart"). There are seven device drivers named:
    - *CCD3:* Driver of main CCD camera
    - *AVT:* Driver of autoguiding camera
    - *FASC:* Driver of main camera focuser (Filter ASCol driver)
    - *FGAS:* Driver of autoguiding camera focuser (Filter autoGuiding AScol driver)
    - *TASC: Driver of telescope and autoguider* (Telescope ASCol driver)
    - *WASA:* Driver of filter wheel A (Wheel AScol A driver)

        - *WASB:* Driver of filter wheel B (Wheel AScol B driver)
        - *WASG:* Driver of filter wheel B (Wheel AScol autoGuider driver)
    - *RPC server is unavailable:* It can be cause by lot of reasons. There is necessary to check internet connection and connection to server machine. If the connection seems to be OK, you can wait until the server will start automatically again (approximately 1 minute) or you can restart the server manually. Another option is to restart RTS-Server manually (type “sudo service rts2 restart”).
- *INFO:* This type of message informs about non-critical events in the system. There are examples of INFO messages:
    - *All Devices are available*
    - *RPC server is available*
    - *Setting send –Camera*
    - *Setting send –Camera binning*
    - *Setting send – Automatic focusing*
    - *Moving start – Focuser*
    - *Moving start – Filter*

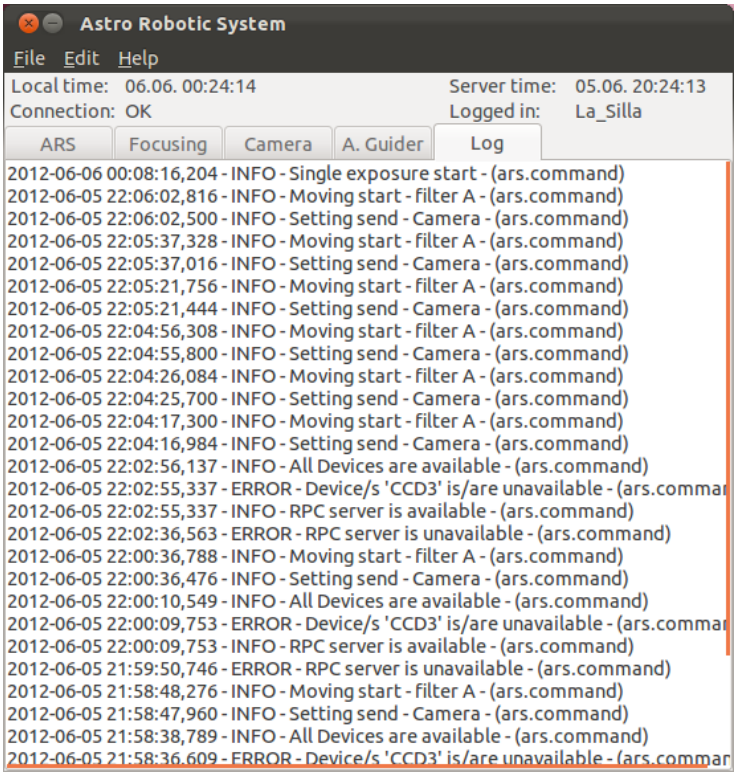

Fig. 13: Log tab

3.3 TROUBLESHOOTING

3.3.1 ERROR - Device/s 'CCD3, CCD3(init)' is/are unavailable

This error message rises in case of problems with controller of main camera or linux driver and entire Linux server could be blocked.

There is necessary to switch off and on a main camera controller power supply. It may be done in “TECHNOLOGIES” screen in TCS visualization (MAIN CAMERA POW. SUP.).

If the problem still remains you have to reboot the Linux PC see 3.3.3 chapter.

3.3.2 Main Camera - BLOCK EXPOSING

Main camera is not able to start exposure in case of “BLOCK EXPOSING” status is active.
“BLOCK EXPOSING” status could appear on main camera tab in ARS-client in these occasions:

- **Telescope** is **slewing**.
- **Filter** wheel of main CCD camera or AG camera is in **undefined position** or the wheel is **moving**.
- **Focuser** of main CCD camera or focuser of AG camera is **moving**.
- **Autoguider** probe is **moving**.

3.3.3 Remote reboot of PC’s

You can use VNC or SSH connection to reboot the PC. If the server is not accessible, you have to use service web server, which works independently on OS.

Open internet browser and type one of these address (you have to be logged into VPN):

Windows (TCS 1) - 134.171.81.76:16992
Windows (TCS 2) - 134.171.81.77:16992
Linux (ARS 1) - 134.171.81.78:16992
Linux (ARS 2) - 134.171.81.79:16992

Use “dk154” as username and appropriate password. Choose “Remote Control” in left column, switch to “Reset” in right field and press “Send Command” button.

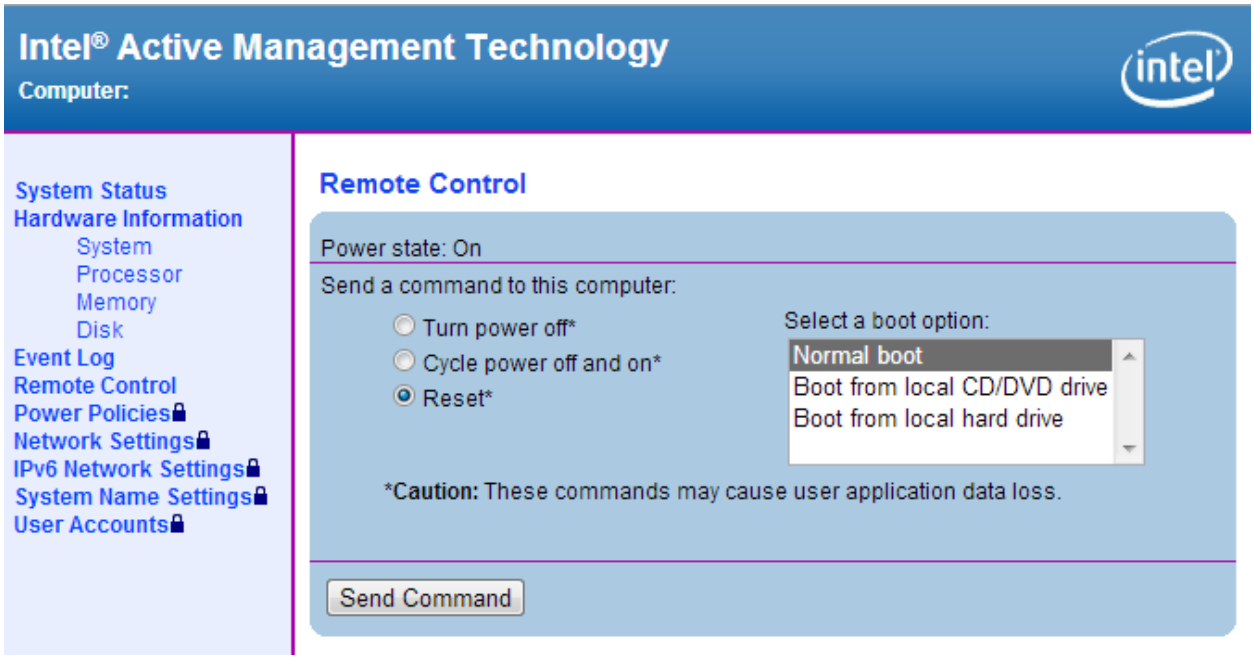

Fig. 14: Service remote control

4 PRIME PARAMETERS

4.1 MAIN CAMERA PARAMETERS

Main CCD camera driver allows sending of initialization commands immediately after driver restart (or after reboot of whole system). These commands are read from script file "/etc/rts2/ccd3commands".

4.2 NAMES OF THE FILTERS

The names of the filters are stored in file "/etc/rts2/devices". There are three filter wheels, each labelled as 'filterd … WASx', where 'x' meaning belongs: A ~ wheel A, B ~ wheel B and G ~ autoguider wheel. The names are located near the end of the appropriate row and enclosed in quotes. Their order corresponds to position of the filter in the wheel. The changes to take effect after restart rts2 service.

5 AUTOMATIC SCRIPT PARAMETERS

ARS includes lot of parameters, which are not usually accessible for users. But sometimes could be necessary to change them. All these parameters are stored in file "/etc/rts2/createvalues". All parameters are restored from this file after rebooting of server.

For example if you change step of automatic focusing, it is necessary to do it in this file as well in order to permanent change (values CCD3_focmode1step, CCD3_focmode2step, AVT_focmode1step, AVT_focmode1step)

Furthermore you could change constant for autoguiding probe positioning (AVT_agsx and AVT_agsx)

All values includes description in the "/etc/rts2/createvalues" file.

6 JSON RPC

Central server is based on RTS2 server. It allows to remote control using JSON RPC. The Java Script Object Notation is a protocol to transmit structured information on text file (or mostly make them available as HTTP pages).

RTS2 API exposes various, both RTS2 devices manipulation, through various calls. The calls are simply URLs, with arguments to parameterize those calls. When the URL is called, retrieved text is in JSON format. This return text contains all data need to.

This section includes list of calls for camera setting, start exposure of main camera, get image data during reading out, start automatic focusing of booth cameras and start autoguiding.

6.1 get

Retrieve values from given device.

6.1.1 Examples

http://localhost:8889/api/get?e=1&d=XMLRPC

http://localhost:8889/api/get?e=1&d=CCD3

http://localhost:8889/api/get?e=1&d=AVT

http://localhost:8889/api/get?e=1&d=FASC

http://localhost:8889/api/get?e=1&d=FGAS

http://localhost:8889/api/get?e=1&d=WASA

http://localhost:8889/api/get?e=1&d=WASB

http://localhost:8889/api/get?e=1&d=WASG

http://localhost:8889/api/get?e=1&d=TASC

6.1.2 Parameters

- ***d*** *Device name. This name has to be equivalent to driver name:*
  - *TASC*
    - *Includes sky coordinates of telescope and position of Autoguider probe*
  - *WASA*
    - *Includes position of filter wheel A*
  - *WASB*
    - *Includes position of filter wheel B*
  - *WASG*
    - *Includes position of autoguider filter wheel*
  - *FASC*
    - *Includes position of main focus*
  - *FGAS*
    - *Includes position of autoguider focus*
  - *CCD3*
    - *Includes values of main CCD*
  - *AVT*
    - *Includes values of autoguider camera*
  - *XMLRPC*
    - *Includes values for parameterize of focusing and autoguiding*
- ***e*** *Extended format. If set to 1, returned structure will containt with values some meta-informations and text description of the value. Default to 0.*

6.1.3 Return

The returned structure is a complex JSON structure, with nested hashes and arrays. Format of the returned data depends on ***e*** *parameter. Simple format, e.g. when e parameter is 0, is following.*

- ***d****:{ data values*
  - *{****value name****:****value****,...}*
- *}*
- ***minmax****:{ list of variables with minimal/maximal allowed value*
  - *{****value name****:[****min****,****max****],...}*
- *}*
- ***idle****:****0|1*** *idle state. 1 if device is idle*
- ***stat****:****device stat*** *full device state*
- ***f****:****time*** *actual time. Can be used in next get query as from parameter*

*When extended format is requested with **e=1**, then instead of returning values in d, array with those members is returned:*

- *[*
- ***flags**, value flags, describing its type,..*
- ***value**,*
- ***isError**, 1 if value has signalled error*
- ***isWarning**, 1 if value has signalled warning*
- ***description** short description of the variable*
- *]*

6.2 set

Set values on server.

*Can set complex values, for example **camera WINDOW** (4 integers)*

6.2.1 Example

http://localhost:8889/api/set?async=0&v=0+2049+100+100&d=CCD3&n=WINDOW

http://localhost:8889/api/set?async=1&v=5.0&d=CCD3&n=exposure

http://localhost:8889/api/set?async=1&v=0&d=CCD3&n=IMAGETYP

http://localhost:8889/api/set?async=1&v=1&d=WASA&n=filter

http://localhost:8889/api/set?async=1&v=1&d=WASB&n=filter

http://localhost:8889/api/set?async=1&v=39.0&d=FASC&n=FOC_DEF

6.2.2 Parameters

- ***d** Device name.*
- ***n** Variable name.*
- ***v** New value.*
- ***async** Asynchronous call. Asynchronous call return before value is confirmed set by the device driver.*

6.2.3 Return

Return values in same format as get call. Return status is 0 if no error occured durring set call, and is obviously available only for non-asynchronous calls.

6.3 expose

Start exposure on given camera.

6.3.1 Example

http://localhost:8889/api/expose?ccd=CCD3&fe=%25N%2FObjectName_%2506u.fits

Please note that example above expect that % and / characters will be properly URI encoded before passing as paremeters to GET call.

6.3.2 Parameters

- **ccd** Name of CCD device. Required.
- ***fe*** *File expansion string incliding „ObjectName*"

6.3.3 Return

Camera values in JSON format.

6.4 runscript

Run script on device. Optionally kill previously running script, or don't call script end, which resets device environment.

6.4.1 Example (CCD camera focusing)

http://localhost:8889/api/runscript?S=exe+%2Fetc%2Frts2%2Fshiftfoc.py&d=CCD3

6.4.2 Example (Autoguider camera focusing)

http://localhost:8889/api/runscript?S=exe+%2Fetc%2Frts2%2Fseqfoc.py&d=AVT

6.4.3 Example (Autoguiding)

http://localhost:8889/api/runscript?S=exe+%2Fetc%2Frts2%2Fguidestar.py&d=AVT

6.4.4 Parameters

- ***d*** *Device name. Device must be CCD/Camera.*
- ***s*** *Script. Please bear in mind that you should URI encode any special characters in the script. Please see* ***man rts2.script*** *for details.*
- ***S*** *Script, but call it without calling script ends (without reseting device state). See s parameter. Only one of the s or S parameters should be provided.*
- ***kill*** *If 1, current script will be killed. Default to 0, which means finish current action on device, and then start new script.*
- ***fe*** *File expansion string. Can include expansion characters. Default to filename expansion string provided in rts2.ini configuration file. Please See man rts2.ini and man rts2 for details about configuration (xmlrpcd/images_name) and expansion characters.*

6.5 killscript

Kill script running on device. Force device to idle, set empty script for it.

6.5.1 Example

http://localhost:8889/api/killscript?d=AVT

http://localhost:8889/api/killscript?d=CCD3

6.5.2 Parameters

- ***d*** *Device name. Device must be CCD/camera.*