

# Instruments remote control project at TNG: SARG implementation

Rosario Cosentino<sup>1,2</sup>, Pietro Bruno<sup>1</sup>, Manuel Gonzalez<sup>2</sup>, Manuel Huertas<sup>2</sup>, Salvatore Scuderi<sup>1</sup>

1 - INAF - Osservatorio Astrofisico di Catania – Italy

2 - Telescopio Nazionale Galileo - Spain

Keywords: Instrument, control, ccd, spectrograph

## ABSTRACT

The high resolution spectrograph at TNG (SARG), mounted at the telescope in 2000, is based on the ‘first generation CCD controller’<sup>1</sup>, transputers and VME real time computer that control the instrument and the detectors. The evolution of the CCD controller, with high performance in speed acquisition and transfer rate, has changed the architecture of the instrument control. Due to the high performances of modern LAN, it has become possible to have direct access to CCD controller and instruments features. The architecture is based on a remote system, connected to a local system through standard network facilities and communicating with it using an XML-like syntax. The remote system receives from the local system commands and, in turn, sends back telemetry and images. This control system will be tested for the first time with the SARG spectrograph, in the framework of the Instruments remote control project at the Telescopio Nazionale Galileo (TNG)

## 1. INTRODUCTION

The general architecture of the ‘SARG remote control’ can be divided into three parts: the spectrograph, the remote computer and the host computers. In this scheme the remote computer is connected directly to the instrument and controls all the features of the spectrograph (CCDs, movements, telemetry), the host computers perform the communication (data and command) between user and remote computer (Figure 1).

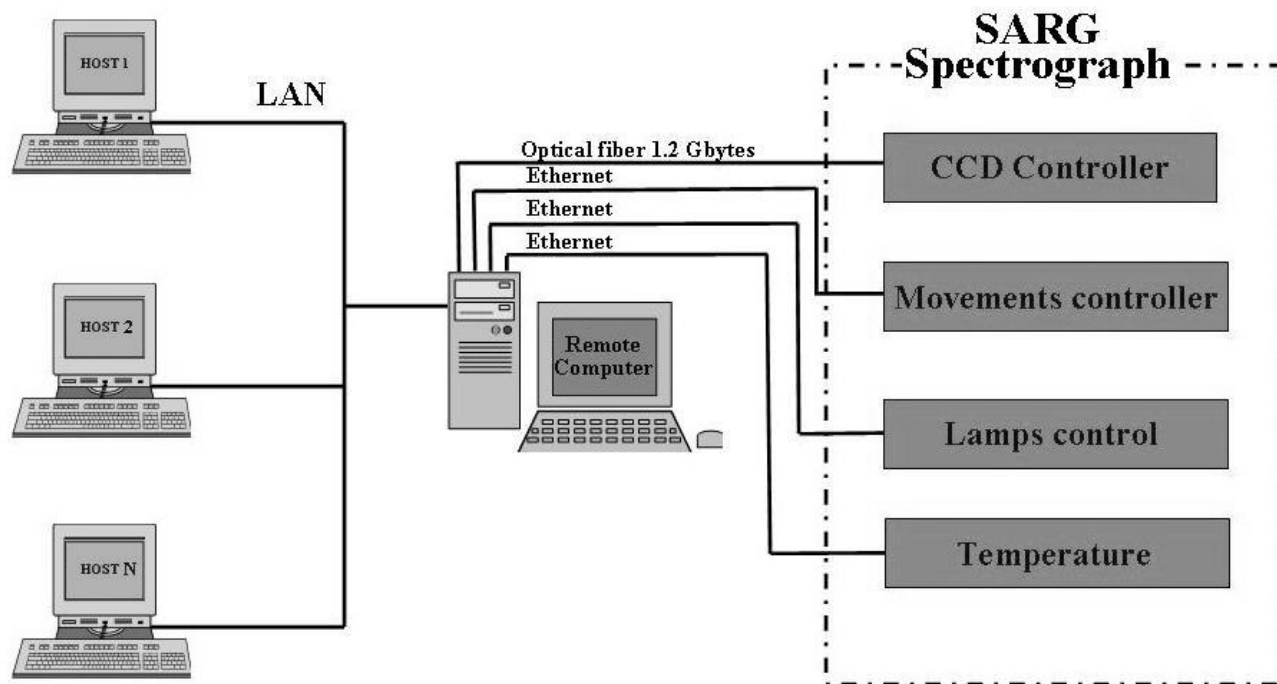


Figure 1 – The general architecture of the SARG remote control system

## 2. THE SARG SPECTROGRAPH

SARG is the high resolution optical spectrograph in operation since late spring 2000 at the Italian Galileo National Telescope (TNG). As one can see from Figure 1 the parts of the spectrograph that needs to be controlled or monitored are the detectors, a series of motorized tables and wheels, the lamps and the temperature sensors. The current SARG control architecture is a VME-based system. This system directly controls 'first generation' CCD controller based on transputers by means of fiber optics link and an Athena board. The instrument functions are instead controlled through R232 ports and stand-alone controllers. In particular, the control of mechanical movements used to select the different operative modes of the spectrograph (grism, filter, lamp selection, polarimeter and field derotation) is obtained by means of ten electro-mechanical motors, servo-controlled by three Owis controllers (mod. DC500); the power control of lamps is performed by a programmable switch based on a CB50 by National Instruments; and, finally, the monitoring of the SARG internal temperature is done by a Lakeshore interface and eight probes with DT470 sensors.

The control software used to manage the instrument is based on tasks running on a VME real-time system and PDOS operating system (Figure 2).

The realization of a 'new generation' CCD controller<sup>2</sup> and the decision to give up the VME-based system to control instrumentation, has brought us to the development of a new control architecture based on a personal computer in a Windows 2000/NT environment. This personal computer, named **remote computer**, controls the entire SARG spectrograph instead of the VME real-time system.

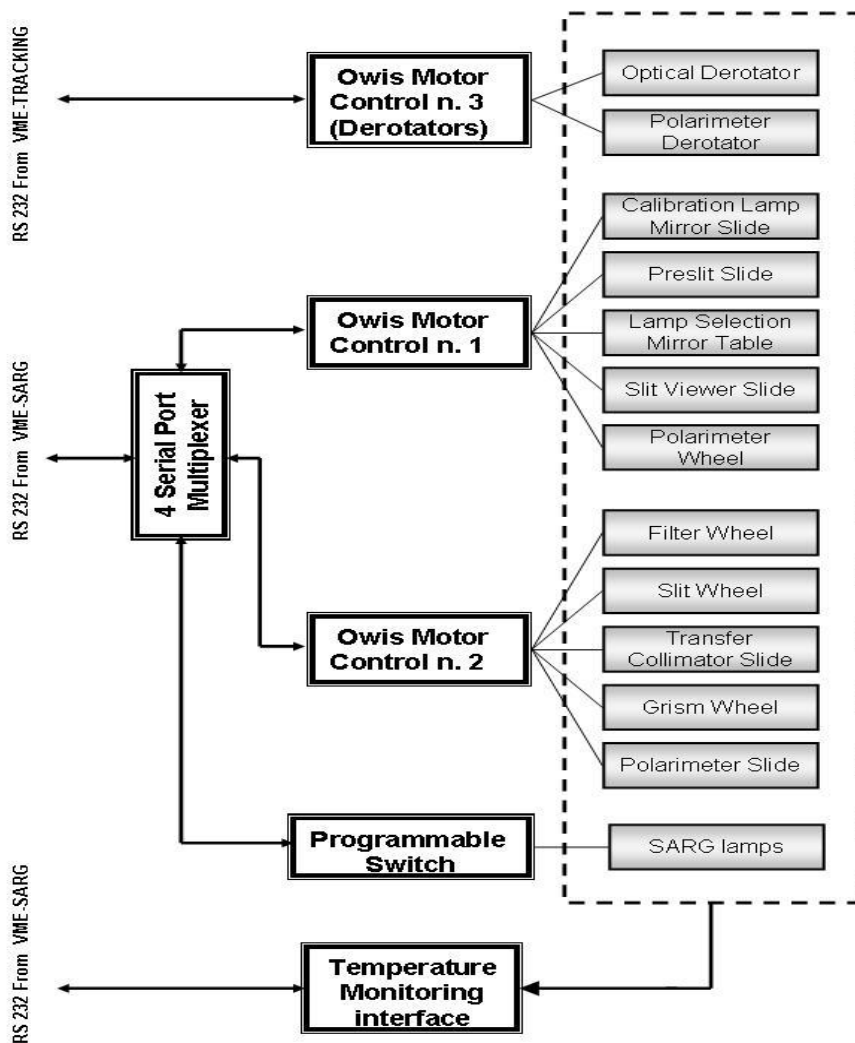


Figure 2 -The current architecture of the SARG instrument control system

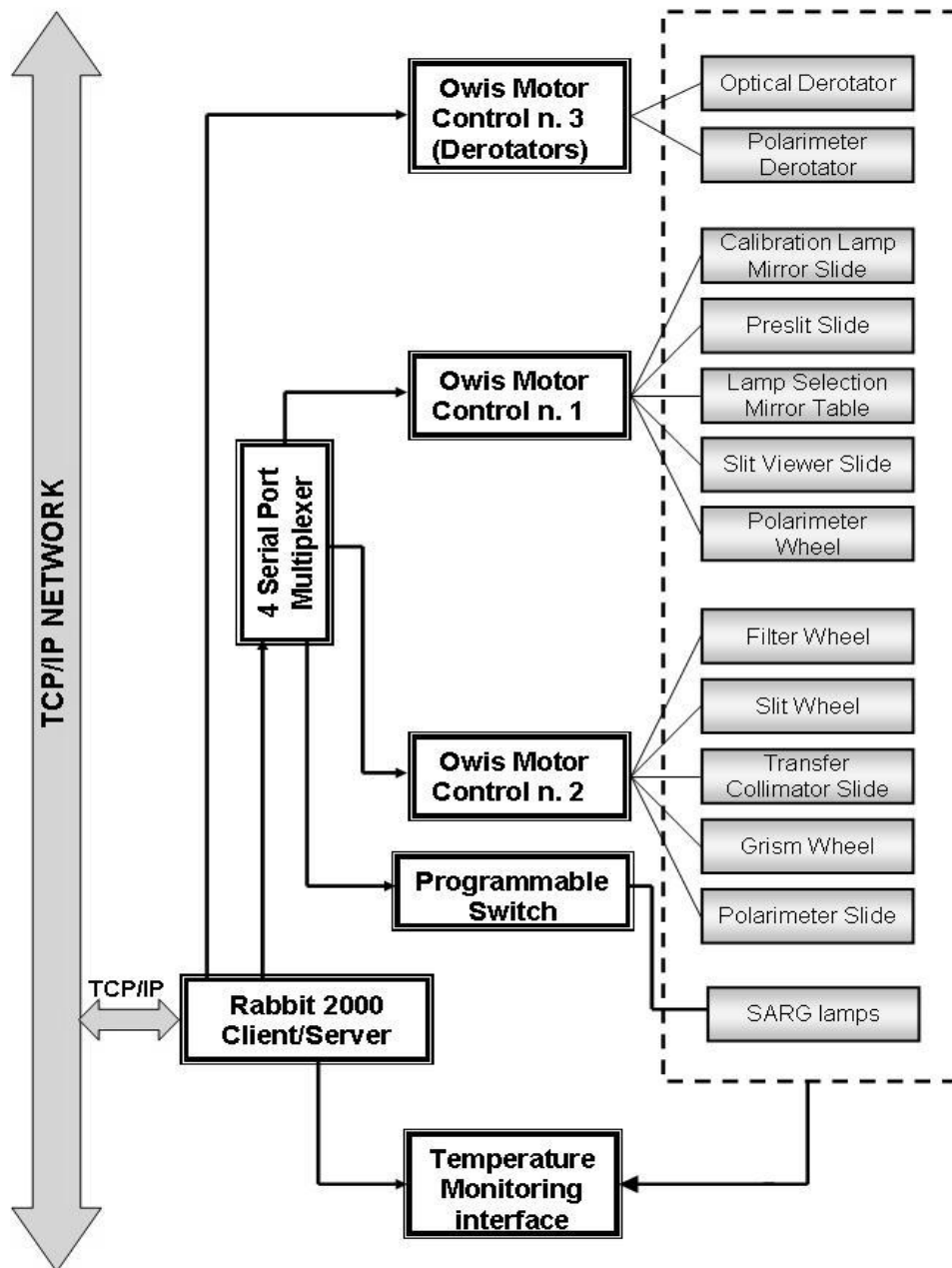


Figure 3 - The new architecture of the instrument control system

### 2.1. The CCD controller

The new generation CCD controller is an evolution of the controller in use at TNG. The unchanged philosophy guarantees a full compatibility with the previous version, in the meantime the technological improvements allow an increase in performance. The upgrade involved the host interface, the communication link, the sequencer, the bias generator, the clock generator, the preamplifier and the signal processing.

The new interface with the host computer is based on a high-speed link and a PCI board and is able to sustain high data transfer rates (1.2 GBoud). The sequencer has been modified to obtain high-speed clocks and different reading modes. A new analogue board, based on fast ADCs and a new signal processing, has been designed. The board is able to process four channels simultaneously, allowing high acquisition rates.

The main differences are a consequence of the change in the architecture of the system. Basically the system can be divided in two parts: the first close to the control computer (local), the second close to the detector (remote). In the new CCD controller the preamplifier is integrated in the analogue board, mounted close to the detector, and the sequence generation is located in the host computer, far from the detector. This is possible thanks to the high speed links capability to serialize and de-serialize the sequence with a 50 ns resolution.

The analogue board is composed by three main parts: the preamplifier, the correlated dual sampler and the bias generator. The preamplifier is based on the low noise operational amplifier THS4061. This configuration allows three gain selection (G-SEL), three band pass filter selection (B-SEL) and programmable offsets.

The Signal Processing is carried out by using the "Correlated Dual Sampling" technique (single sampling is also allowed). The output offset can be adjusted through a 14-bit D/A Converter. The A/D Converter is the LTC1608, that has a 16 bits resolution and a 2  $\mu$ s conversion time.

In summary, the analogue board allows three selectable input gains and three selectable bandwidths to work at different readout speeds, the possibility to program the offsets before and after the CDS stage, to adapt the signal to the A/D converter and the choice of different references for the dummy input. The bias generator allows 16 programmable bias voltages with different ranges, divided in four groups. The voltage ranges are: 15÷ 30 Volts, 5÷ 15Volts, -5÷ 5 Volts and -10÷ 10 Volts. The performances of the new controller have been tested in laboratory. The acquisition rate has been increased up to 7 times with respect to the old system without any degradation of the readout noise.

## 2.2. Movements and lamps control

The new generation of the software has been developed for Windows 2000 environment, in particular two DLLs (Dynamic Link Library) have been developed in C language; they communicate with the hardware, through socket with a system based on Rabbit 2000 that converts TCP/IP protocol into RS232 protocol.

## 3. REMOTE COMPUTER

The remote computer hardware (Figure 4) is composed by a personal computer on a windows 2000 platform, equipped with a PCI board (that communicates with the CCD controller), an Ethernet board and a two levels software:

- A server program
- A low level software

The server program is based on a TCP/IP socket and can receive commands from a well known IP address and port, and after executing them it reports the result to the client program originator of the request. Several commands can be sent to the server at once, since it implements an internal queue. The server program communicate with the underling hardware, CCD and instrument controllers, using *ad hoc* DLLs. These DLLs contain all the necessary functions for the management of detectors and movements. In addition, the server program communicate with the client using a XML-like protocol in which the commands, with their parameters, are sent to the server. The server responds also in a XML-like manner after the completion of the command.

The low level software may be divided into two parts: a first one hardware-dependent and a second, accessible by the server program. In this application the first is a driver that controls the PCI board, used for the low level communication with the CCD controller, and the second is a DLL (Dynamic Link Library), that allows the high-level control. For the movements control the same method is used: there is a DLL library, that allows the remote control of the SARG movements through a socket and a dedicated communication hardware, based on Rabbit modules.

In summary, the micro-command executed by the server program, executes a series of DLL functions and each function manages the hardware through a driver or socket communications.

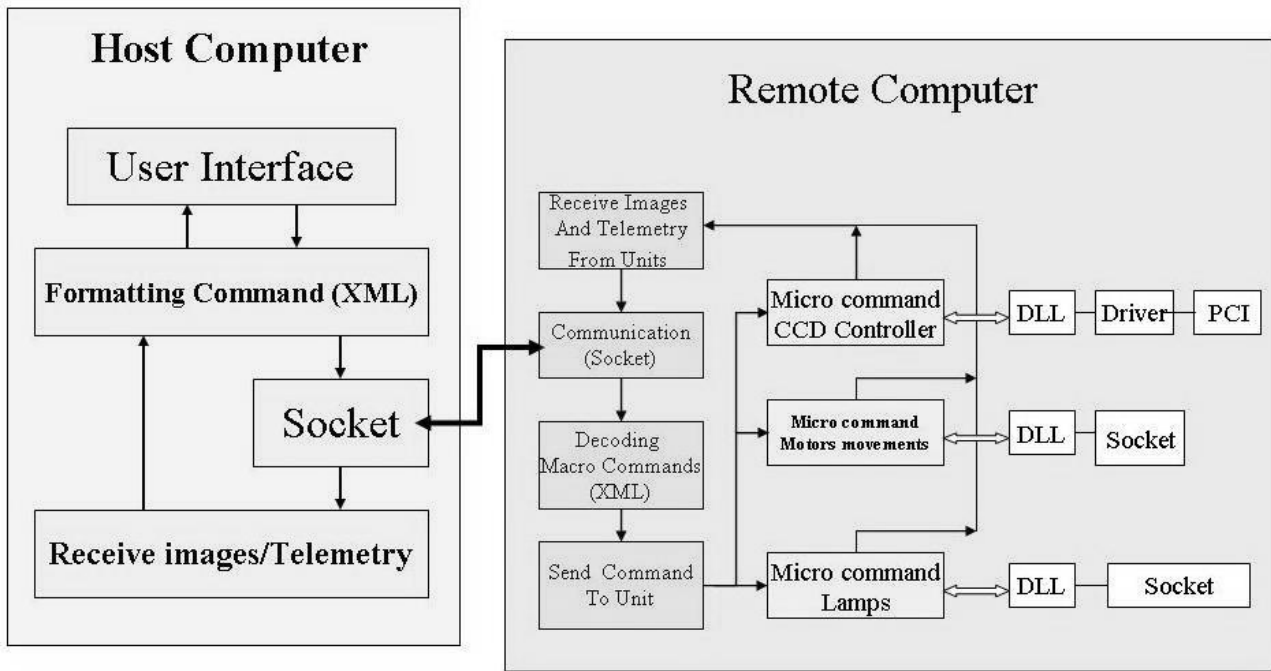


Figure 4 - Host-Remote connection

### 3.1. Remote-SARG connection

In order to communicate with the SARG hardware, the remote computer uses two kinds of connection: an optical link to the CCD controller and the LAN to the instrument controls (Figure 5).

The module that manages commands dispatching from LAN to RS232 is a Rabbit 2000 module, that converts the TCP/IP to RS232 protocol. Furthermore, the Rabbit module is able to manage the Power control of lamps and instrument modules.

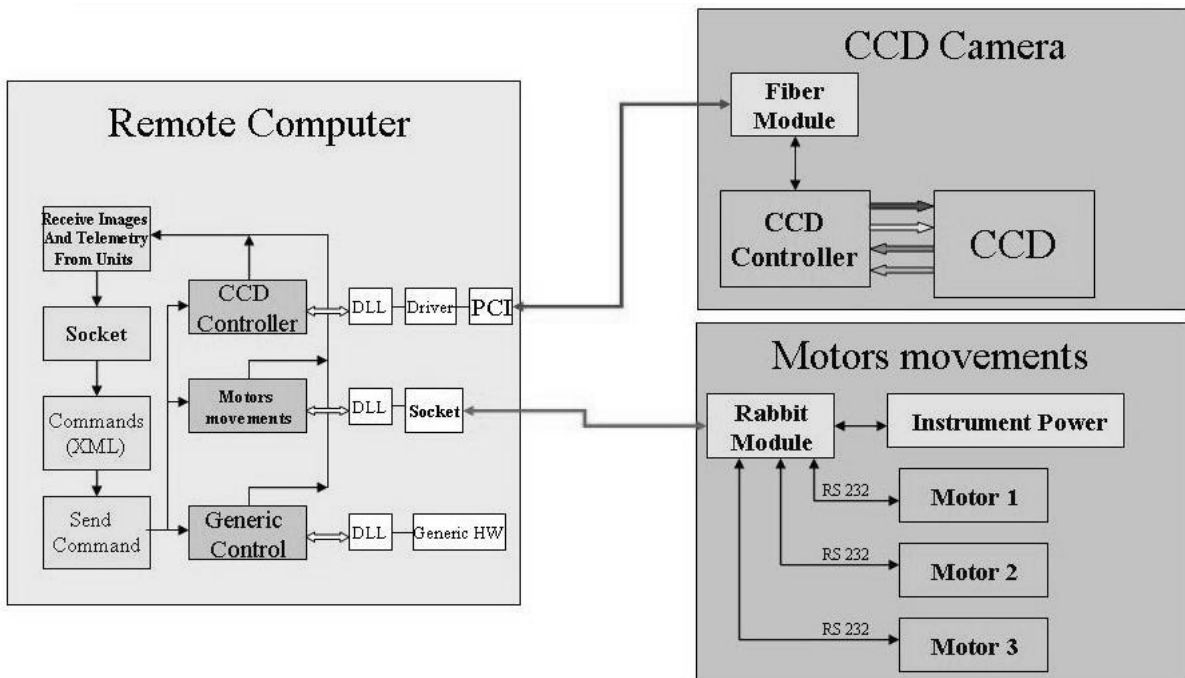


Figure 5 - Remote-SARG connection

## 4. HOST COMPUTER

The host computer can be a nonspecific platform, equipped with an Ethernet board able to communicate through a TCP/IP client socket with the remote computer.

In this system the communication is based on socket connections and XML-like-commands. The user interface formats the command or the image/telemetry request into XML-like-commands and sends it to the remote computer. The remote computer, receiving the XML-like-command, replies with a 'command received' XML-like-string. After the command execution, the remote computer sends a XML-like-executed string with the error code to the host computer.

### 4.1. The Graphic User Interface

The core of the graphic user interface (GUI) will be written using IDL language. IDL allows to combine very powerful tools for data analysis with simple tools to construct and manipulate graphical interfaces using widgets and object-oriented programming. To communicate with the remote computer the GUI will use the IDL-Java bridge, which allows to access Java object within the IDL code. The communication will be realized through Java class sockets. There will be two sockets, one to send commands and receive telemetry, errors and messages and another one to transfer CCD images. The user interface will exchange messages, commands and telemetry with the remote computer using an XML-like syntax. The GUI software has the advantage to be independent from hardware technological upgrading, in fact, every change in the instrument is restricted to the remote computer and does not affect the host computer.

### 4.2. The communication protocol

The eXtended Markup Language (XML), is a human-readable and machine-understandable way to describe structured data. We are using XML-like syntax to describe command sets (including parameters, data-types and constraints) and their associated formats, telemetry, and communication mechanisms.

When the server is started, it creates two listening sockets. One for the both commands and telemetry and another for the images. Once the client connect to the server, the server is ready to accept commands from the client and, in case, send the image as result of the exposition back to the client.

To respond to our needs we have defined the following XML-like structures

- **Command structure**

The structure which starts the execution of a command on the server side has the following format:

```
<CMD>
  <ID>...</ID>
  <ACRONYM>...</ACRONYM>
  <OPERANDS>
    <OP0>...</OP>
    <OP1>...</OP1>
    ...
  </OPERANDS>
</CMD>
```

The ACRONYM identifies the command to be executed. The ID must be a different and unique number for every request and it is used to identify, when a response is sent back to the client from the server, to which command is associated the response. The operands are optional meaning that a command can have different number of operands or no operand at all.

- **Submitted command structure**

Once a command is received for the server, a submitted command structure is sent back to the client, acting as an acknowledge. This structure means only that the command has been received and it has been included in the queue, to be executed as soon as possible.

- **Executed command structure**

A different structure is sent to let the client know that a command has been executed. This structure contains an ID parameter used to identify which command has completed its execution, and a VAL parameter that contains an exit code. If the execution of the command was successfully completed, this code is equal to one, otherwise it contains the error code to extract the reason why the command has failed.

- **Error structure**

Whenever an error happens, which is not related to the execution of a command, a different structure is sent. This structure contains a CODE parameter and a MSG parameter with a short description of the error.

- **Telemetry structure**

If the command sent to the server was asking for telemetry, it sends back to the client a telemetry structure which contains an ACRONYM parameter identifying the telemetry variable and a VALUE parameter containing the corresponding value.

- **Image structure**

If the command sent by the client was an exposure command, when the exposure is over and the CCD has been read-out the server sends the CCD image back to the client using a different socket. The image structure contains a minimal header with the information necessary to reconstruct the image on the client side. In particular, the HEADER element contains an ID sub-element which refers to the expose command which generated the image and the sub-elements DIMX and DIMY with the image dimensions. The BYTES element contains the CCD image in binary format. .

#### 4.3. Code Example

In this paragraph one example shows the use of two different types of XML-like structure: command (CMD) and images (IMG).

For instance, if you need a dark exposure of 60 seconds, you need to specify exposure time and image type so the XML-like command structure would look like this:

```
<CMD>
  <ID>1</ID>
  <ACRONYM>EXPOSE</ACRONYM>
  <OPERANDS>
    <OP1>60</OP1>
    <OP2>dark</OP2>
    <OP3>0</OP3>
  </OPERANDS>
</CMD>
```

The 1024×2048 pixel image structure returned by the remote computer is:

```
<IMG>
  <HEADER>
    <ID>1</ID>
    <DIMX>1024</DIMX>
    <DIMY>2048</DIMY>
  </HEADER>
  <BYTES>
    binary image
  </BYTES>
</IMG>
```

## 5. INTEGRATION IN THE TNG CONTROL SYSTEM

The TNG control system is based on a series of control workstations, integrated in the TNG local area network, connected to VME crates that constitute the telescope subsystems and/or instruments. WSS (Workstation Software System) is the software that manages the communication between each workstation and the low level control software constituted by GATE (GALILEO Telescope Environment). GATE runs on VME crates over the PDOS real-time operating system.

To exchange information with the TNG system the host computer needs to be integrated in WSS. The GUI will manage the communication with WSS (see Figure 6) using *ad hoc* C libraries. This will allow to pass to WSS and then to the archive the instrument configuration and the CCD images. Furthermore the host computer will be able to retrieve information from the TNG environment (for example telescope position).

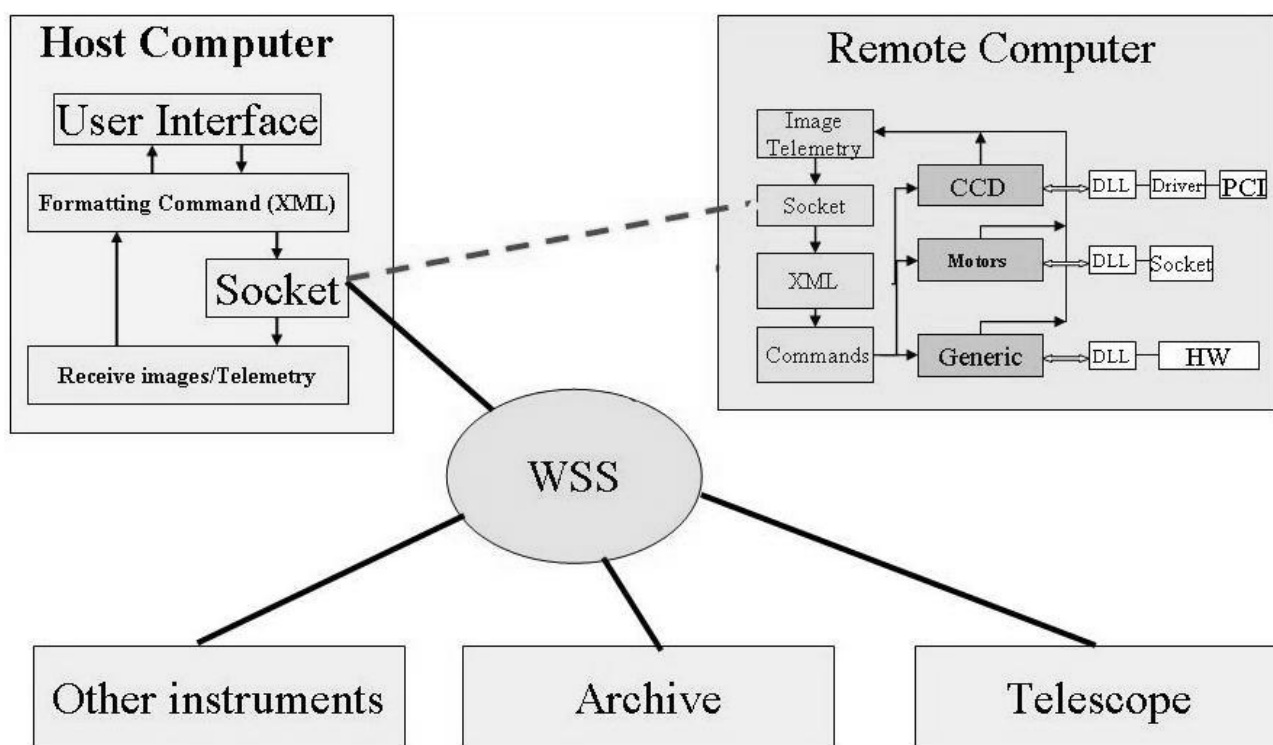


Figure 6 - Remote instrument control system in TNG architecture

## 5. CONCLUSION

The development of new CCD controllers and the upgrading of the TNG control system has driven to the design of a new architecture for the remote control of part of the TNG astronomical instrumentation. We have described the implementation of this control system applied to the case of the SARG spectrograph. Its main characteristics can be summarized as follows:

- Remote management of the instrument (power control, motorized wheels and tables, light sources, temperature sensors) through the network by means of Rabbit 2000 modules.
- Communication between host and remote computer through sockets.
- Separation of low level software from graphical user interface. This makes hardware upgrade on the remote control computer and on the instrument independent from the host computer.
- Platform independence of the GUI. This is obtained through the use of IDL and of a communication protocol based on XML-like structures. This characteristics allows the assignment of processes to different remote computers regardless of their operating system.
- Extendable and flexible architecture which can support a wide variety of astronomical instruments.
- Easy integration in the TNG control system based on WSS.



## REFERENCES

1. Bortoletto, F., Bonoli, C., D'Alessandro, M., Fantinel, D., Farisato, G., Bonanno, G., Bruno, P., Cosentino, R., Bregoli, G., Comari, M., *CCD cameras for the Italian National Telescope Galileo*, SPIE Proceedings Vol. 2654, January 1996
2. F. Bortoletto, M. D'Alessandro, D. Fantinel, E. Giro, L. Corcione, G. Bonanno, P. Bruno, Cosentino, A. Carbone, G. Evola, *A new generation of detector controllers*, Memorie della Societa Astronomica Italiana, v.74, p.159 (2003) - 2003MmSAI..74..159B
3. R. Cosentino, M. Belluso, G. Bonanno, P. Bruno, F. Bortoletto, M. D'Alessandro, D. Fantinel, E. Giro, L. Corcione, A. Carbone, G. Evola, *The new generation CCD controller: first results*, Scientific Detectors for Astronomy - June 16-22, 2002 - Waimea, Hawaii
4. R.G. Gratton, G. Bonanno, P. Bruno, A. Cali, R.U. Claudi, R. Cosentino, S. Desidera, F. Diego, G. Farisato, G. Martorana, M. Rebeschini, S. Scuderi, *Sarg: The high resolution spectrograph of TNG*, Experimental Astronomy – Kluwer Academic Publishers (EXPA459)
5. Balestra A., Marcucci, P., Pucillo M., Vuerli, C, *The GALILEO Project. Workstation Software System and User Interface*, Progress in Telescope and Instrumentation Technologies, ESO Conference and Workshop Proceedings, 1992ptit.conf..761B