RevMexAA (Serie de Conferencias), **45**, 30–31 (2014)

III Workshop on Robotic Autonomous Observatories (Torremolinos, Malaga, Spain, 7-11 October 2013)
Editors: J. C. Tello, A. Riva, D. Hiriart & A. J. Castro-Tirado

ATA50 TELESCOPE: SOFTWARE

B. B. Güçsav,[1,2] Y. Kiliç,[1] and M. N. Shameoni[1]

RESUMEN

Buscamos implementar una estructura de observatorio completamente autónomo para el telescopio ATA50. Todos los componentes del observatorio están actualmente controlados manualmente por aplicaciones de MS Windows. Un sistema OCS bien diseñado, como RTS2, puede proveernos una solución rápida. GNU/Linux es nuestro sistema operativo predilecto al considerar la filosofía código abierto y el desarrollo de RTS2 (Kubanek et al. 2006). Como era de esperarse, hemos encontrado problemas con drivers ausentes/dolorosos para nuestro equipo. Aceptar estas condiciones ha hecho que codificar los drivers de componentes físicos constituya un enorme y complejo OCS.

ABSTRACT

We aim to implement a fully autonomous observatory structure for ATA50 telescope. All components of the observatory are currently controlled by manual MS Windows applications. A well designed OCS such as RTS2 might provide us a quick solution. GNU/Linux is our choice for the OS by considering the open-source philosophy and the original development environment of RTS2 (Kubanek et al. 2006). As expected, we have faced absent/painful driver issues for our equipment. The compromise for the above has made for coding the physical component drivers regarding development of a huge and complex OCS.

Key Words: methods: data analysis — techniques: image processing — instrumentation: miscellaneous — telescopes

1. INTRODUCTION

The success in this project, hopefully makes us to attain -at least- some of the following goals;

- A telescope network that can automatically run a pool of observations, increases the overall efficiency. The realisation of such an autonomous observatory may constitute an initiative for a national telescope network (i.e. "high availability" network).
- Today, autonomous observatories are inevitable for GRB and various transient observations. Globally distributed telescope networks have proved to be successful for these type of observations. It is highly desirable to be a participant of such an international telescope network.
- Turkey is going to have a middle class (4 meter) telescope in DAG[3]. We think that, the experience that will be gained, even in such a -relatively- small autonomy project is invaluable.

So, this is just the beginning of a struggle...

2. ATA50 CURRENT SOFTWARE INFRASTRUCTURE

The Alluna Optics OTA, ASA DDM-160 mount, FLI filter wheel and Apogee Alta CCD camera can be controlled under the MaximDL, besides, the mount and OTA have manufacturer supplied control applications. The custom made rotating roof is operated manually from a PIC controller with the help of a simple VB application (A PLC system is going to be applied soon).

The observatory network infrastructure consists of a shared 1 Gbps fiber connection to the operations building. The observatory network is in a shared VLAN provided by the university campus CC. Thus, the privacy needed is obtained via a Layer-3 VPN (openvpn). In addition to the fiber connection, an in-case-of-emergency wireless radio link access is planned to be deployed very soon.

3. ONGOING EFFORTS / RTS2 IMPLEMENTATION

Information obtained regarding the mount until now:

- The mount is controlled via usb to serial interconnect (FTDI 232LR usb serial converter.) The DSP chip on the control board is a TI 2808PZA.
- We had conjectured three possible ways to code the mount driver; Writing a driver code from

[1]Astrofizik Araştirma ve Uygulama Merkezi, Atatürk Üniversitesi, 25240 Yakutiye, Erzurum (atasam@atauni.edu.tr).
[2]Astronomi ve Uzay Bilimleri Bölümü, Ankara Üniversitesi, 06100 Tandoğan, Ankara.
[3]Eastern Anatolia Observatory, `http://dag-tr.org`.

III Workshop on Robotic Autonomous Observatories (Torremolinos, Malaga, Spain, 7-11 October 2013)
Editors: J. C. Tello, A. Riva, D. Hiriart & A. J. Castro-Tirado

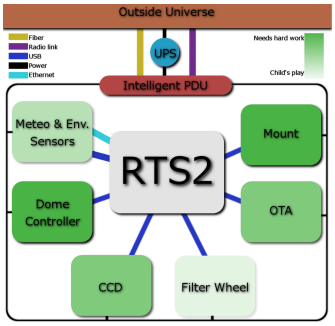

Fig. 1. Estimated implementation difficulties.

scratch with Labview (after a quick research, it is turned out to be the least possible approach due to the lack of Labview support to TI 2XXX series DSPs), reverse engineering by usb sniffing (by all possible means with hardware or software sniffers), decompiling and disassembling the current Windows drivers (including .NET and pre .NET DLLs) from the manufacturer(complying with licence agreement).

Implemented RTS2 steps up to day (Fig. 1):

- The only device that is successfully controlled under RTS2 is FLI filter wheel. The Apogee-Alta U230 camera has been used with the drivers and the camera application provided by The Random Factory. We, however, have not been able to integrate into RTS2 yet. We achieved to make a virtual observation (all of the RTS2 daemons but the filter wheel were dummy).

TODO: Integration of various environmental devices including meteo and surveillance systems. Observatory has a Davis Vantage Pro 2 meteo station and the DAG project has currently access to EUMETSAT. The meteorological data coming from the satellite has been processed through a pipeline (TMet[4] and various utilities coded by our team) on a daily basis. We are planning to bring them to the usage of RTS2 OCS.

4. CURRENT CODEBASE : MYRAF AND EXPERIENCES FROM ROTSEIIID

We have used a small cluster of commodity PCs (total of 40 cores) for our previous project for ROTSEIIId (Akerlof C. W. et al. 2003) archival data. In order to process bulk amount of photometrical data (∼400K frames) obtained from ROTSEIIId archive, we coded a small parallel application (Güçsav, B. B. et al. 2012) using MPICH libraries that can also perform multi differential photometry (Fig. 2). A

```
1         void *
2         HOG_MPI_Gather_v3(void *sendbuf_a, int root_id){
3            void    *recvbuf_a;
4            int     i,j,k;
5                  recvbuf_a=(struct HOG_struct *)calloc(
                        _TOTAL_FITS_FILES_ * (_CS_END_A_[
                        root_id]          _CS_START_A_[root_id
                        ]),  sizeof(struct HOG_struct));
6
7                  _RECVDISPS_A_[0]=0;
8         for(i=0;i<_N_PROCSIA_;i++){
9                        _RECVCOUNTS_A_[i]=_JOBS_PER_CORE_A_
                              [i]*(_CS_END_A_[root_id]-
                              _CS_START_A_[root_id]);
10                           if(i>0)_RECVDISPS_A_[i]=
                                    _RECVCOUNTS_A_[i-1]+
                                    _RECVDISPS_A_[i-1];}
11
12                 MPI_Gatherv(sendbuf_a, _JOBS_PER_CORE_A_[
                        _CL_ID_]*(_CS_END_A_[root_id]-
13                         _CS_START_A_[root_id]),
                              MPI_HOG_struct, recvbuf_a,
                              _RECVCOUNTS_A_, _RECVDISPS_A_,
14                         MPI_HOG_struct, root_id,
                              MPI_COMM_WORLD);
15                 return recvbuf_a;}
```

Fig. 2. A piece of humble parallel code.

number of shell scripts (using miracles of E. Bertin[5], i.e. Sextractor, SCAMP et.al.) run before the parallel code takes action. MYRaf[6] is an open source (GPLv3) fast and reliable photometrical astronomy data analysis software written in Python. Qt is the graphical framework, PyRAF is the glue of MYRaf. It can be used in manual (GUI) or automatic mode. MYRaf's (v1.5 Beta) current abilities are, prereduction, source registration (with aligning) and automatic photometry of multiple objects in multiple frames simultaneously. Future goals of MYRaf are; Ability to make NEO photometry, WCS implementation and VO standardization, and data archive management.

The experience gained from our codebase and the success of RTS2 encourage us to dream of a full pipeline that turns the raw image data to scientifically meaningful results automatically.

Acknowledgements Funding for the ATA50 Telescope and all the infrastructure was provided by Atatürk University Scientific Research Projects Programme (GBAP 2010/40, BAP 2011/373, BAP 2011/374, BAP 2011/366, BAP 2012/497). This study is also supported by Ministry of Development Project with the project name "Doğu Anadolu Gözlemevi - DAG" and the project number "2011K120230".

REFERENCES

Kubánek, P., Jelìnek, M., Vítek, S., et al. 2006, SPIE, 6274, 59

Akerlof C. W., Kehoe R. L., McKay T. A., et al., 2003, PASP, 115, 132

Güçsav, B. B., Yeşilyaprak C., Yerli S. K., et al., 2012, Exp. Astron., 33, 1, 197

[4]TMet Project, `http://tmetproject.org/`.

[5]E. Bertin, `http://astromatic.net/`.

[6]MYRaf Project, `http://myrafproject.org/`.