Proceedings of the VII Bulgarian-Serbian Astronomical Conference (VII BSAC)
Chepelare, Bulgaria, June 1-4, 2010,
Editors: M. K. Tsvetkov, M. S. Dimitrijević, K. Tsvetkova, O. Kounchev, Ž. Mijajlović
Publl. Astron. Soc. "Rudjer Bošković" No 11, 2012, 219-228

A DISTRIBUTED COMPUTER SYSTEM CONCEPT OF THE ASTROINFORMATICS PROJECT

A. KOLEV[1], D. KALAGLARSKY[2], D. DIMOV[3], M. TSVETKOV[4]

[1]*National Military Academy G. S. Rakovsky,*
Lulin Distr., Block 608-D, fl.8, 1336 Sofia, Bulgaria,
E-mail: alexkolev@yahoo.com
[2]*Institute of Mathematics and Informatics, Bulgarian Academy of Sciences,*
8 Acad. Georgi Bonchev Str., Sofia 1113
E-mail: damyan@skyarchive.org
[3]*Institute of Information Technologies, Bulgarian Academy of Sciences,*
Acad. G. Bontchev Str., Block 29-A
1113 Sofia, Bulgaria
E-mail: dtdim@iinf.bas.bg
[4]*Institute of Astronomy, Bulgarian Academy of Sciences,*
72, Tsarigradsko Shose Blvd., 1784 Sofia, Bulgaria
E-mail: milcho@skyarchive.org

Abstract. The paper points to building of a R&D environment for the image processing of astronomical wide plates. Appropriate topics have been generally discussed like common environment architecture, scientific background of information storage, software tools for results presentation and usable fragments of software program code.

1. INTRODUCTION

The Astroinformatics' project[1] opens a wide range of separate problems in the cross border area of mathematics, astronomy, and informatics. The project general objective is digital preservation of the great amount of astronomical wide plate images collected in the years either in the country or world wide. The preserved astronomical images, we call them astro-images, should be well usable via Internet in the frames of a Virtual Observatory (Kounchev et al., 2009). Except the overall testing environment this paper discusses also many specific problems of the

[1] The present research has been conducted in the frames of Astroinformatics project supported by the National Science Fund at Bulgarian Ministry of Education & Science (grant # DO-02-275/2008).

intended distributed system for keeping and processing the collected astro-images. The structure of Astronomical Database (ADB) for astro-images is considered of main interest hereinafter. A specific viewpoint is put on the FITS file format that is wide spread for transferring of astro-images among systems and/or end-users (CFITSIO, 2009). In this line of thinking we also describe the practical usage of input/output software routines to obtain and modify tabular and graphics content of ADB. For the project presentation purposes ADB should also have a part for geo-referred data for the astronomy observatories around the world. ADB will also need the so called AstroWeb subsystem to be designed supporting the project specific data/information structures and presenting them in a web environment. Another specific problem of ADB is organization of the very large data flows between separate work places in distributed computer network of the Astroinformatics system we intend.

In this paper we describe the web-based geoinformation system proposed in the frames of the Astroinformatics project, stressing on the testing environment and user interfaces.

2. THE TEST ENVIRONMENT DIAGRAM

The main modules of testing environment proposed are shown in Fig. 1 with all its items explained as follows.

- **Remote Common Database** – still under construction it is designed for common usage of all participants of the project. It consists of 3 basic parts (sub-DBs). **Firebird RDBMS** – a well known database system, licensed under GPL (General Public License) that is already used to support the current WFPD (Wide Field Plate Database), to this end – observatory catalogs and pictures of WFP type. **FITS astro-database** – a part of ADB to be designed for keeping WFD astro-images given in FITS (Flexible Image Transport System) format. **Geodatabase** – consists of geographical data (geo-referred positions) of important objects, to this end – the astronomical observatories related to WFPD.
- **Application Server** – a high productive computer configuration of „server” type intended for astro-images processing. The respective software modules will be implemented herein as binary code and be directly managed by a web interface. The end-users will obtain definite information services via the respective web browser at their work places.
- **Web Server** – a high productive computer configuration with an IIS (Internet Information Services) or Apache system installed. It will support the web access information services exported by the Application Server.

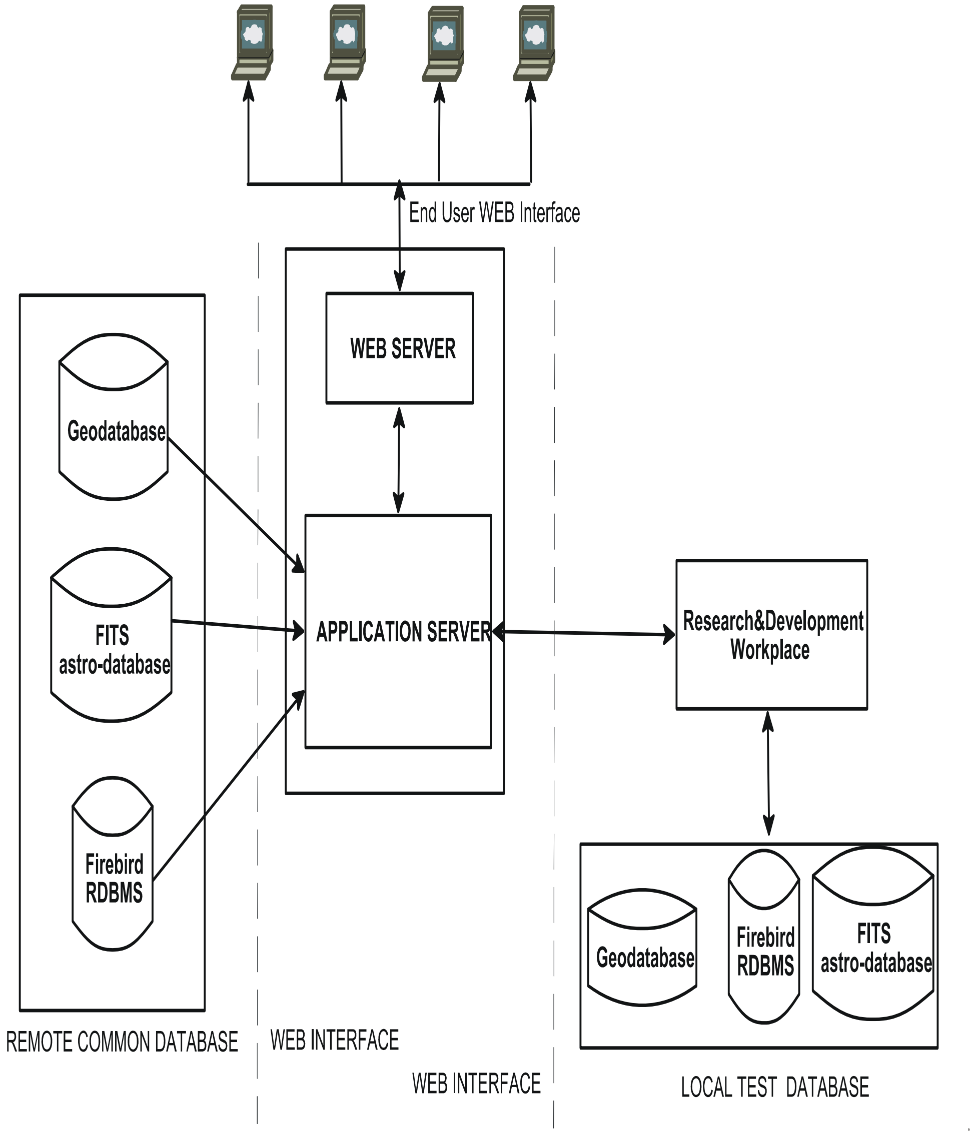

Figure 1. The main modules of testing environment.

- **Research & Development Workplace –** one or many workplaces (local or remote ones) designed for development and test of image processing modules. Many well known modules in astro-image processing practice will be set up and tested. Development of specific (original) modules is also intended by necessity. A stress will be given on modules supporting FITS format directly. Each workplace is considered to be equipped by a software development environment specific to the respective user and/or developer.

- **Local Test Database –** it is a local database for set up and test purposes at the application server place. It represents a test subset of the main database to be accessed only by the R&D workplaces but not by the user ones.
- **End User Web Interface –** it represents a web page delivering an user-friendly interface to several information services of Astroinformatics specifics.

3. WIELD FIELD PLATE DATABASE – A BRIEF PREVIEW

The Wide-Field Plate Database (WFPDB) (Tsvetkov et al., 2009) contains descriptive information for the astronomical wide-field photographic observations stored in numerous archives all over the world. It is capable to provide on-line access to the data for about more than 2 million of observations from nearly 500 archives obtained since the end of last century. Presently the WFPDB includes data for about 600 000 observations from 300 plate catalogues.

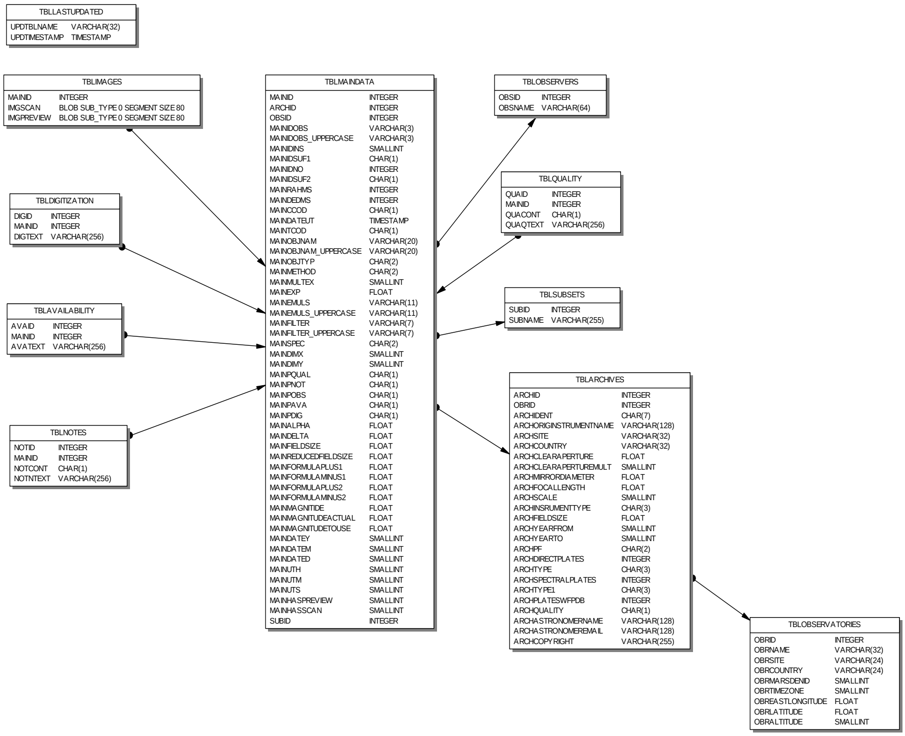

Figure 2. Tables from WFPDB.

The WFPDB provides for each observation a plenty of information for the corresponding archive, for the parameters of the observation instrument, for the observation parameters (position on the sky, observation time, object name, method, exposure time, emulsion type, filter type, spectral band, plate size), as well as data for the plate quality, for the observers and their comments written on/about the plates.

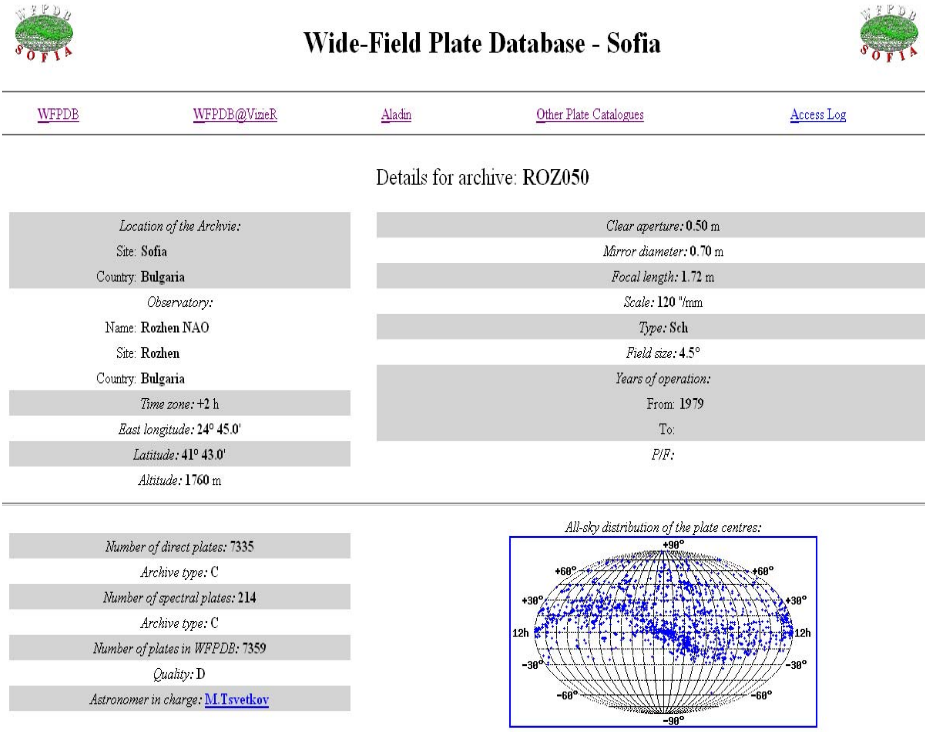

Figure 3. Details from archive.

A web based search engine was developed that may be assessed on web space by a properly user interface (Kolev, 2009). The WFPDB logical schema is illustrated in Fig. 2. The schema is generated using the reverse engineering tools (IB Expert 2004). A screenshot from currently used WFPDB is explained in Fig. 3.

There are envisaged several possible manners to user-friendly exploration of the large set of astronomical data in WFPDB. One of them is to select an observatory, then extract the subset of archives according to this observatory and at last get a limited list of astro-images. Each point in plate centers diagram corresponds to stored astro-image as shown in Fig. 4.

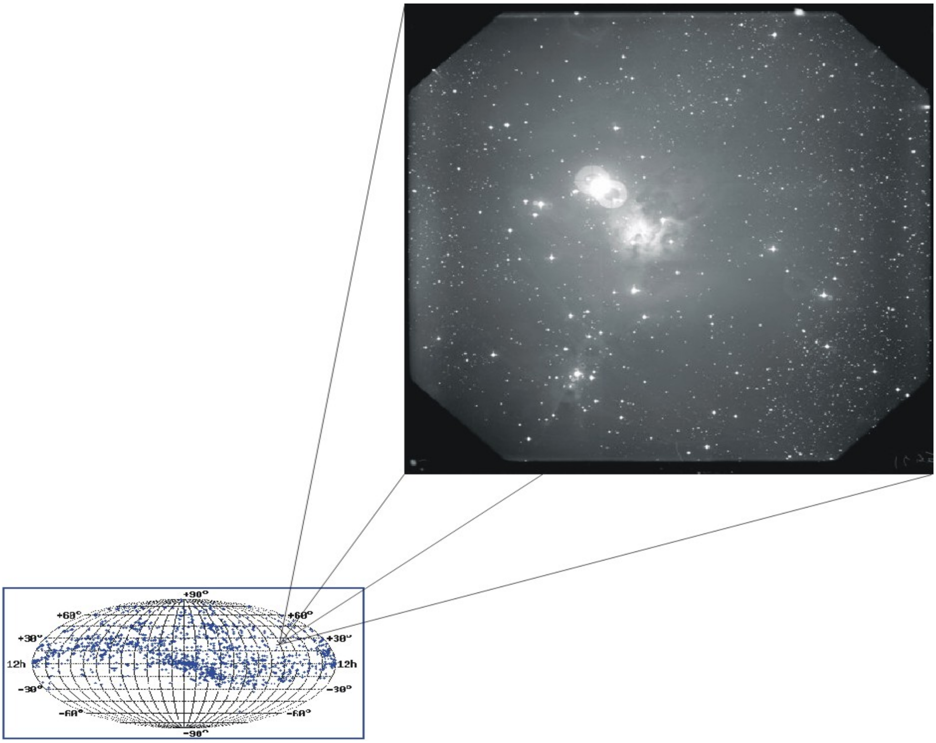

Figure 4. Plate center as astro-image preview.

4. FITS IMAGE ASTRO-DATABASE SOFTWARE ACCESS

The acronym FITS stands for Flexible Image Transport System. The name itself expresses the general purpose of FITS to be a flexible mean for transferring image data (and extra information about them) between cooperating computer systems.

For astro-image processing purposes, where FITS file format is predominantly used, a specific software access to file data is necessary to be implemented. CFITSIO is a library of C and FORTRAN subroutines to read and write data files in FITS format (CFITSIO, 2009). CFITSIO is developed by the High Energy Astrophysics Science Archive Research Center (HEASARC). It provides simple high-level functions/routines for reading and writing of FITS files in this way insulating the programmer from the internal complexities of the FITS format. CFITSIO also gives many advanced opportunities for manipulate and/or filter the FITS information. CFITSIO is already tested on many operating systems and software platforms, most usable of which are:

OPERATING SYSTEM	COMPILER
HP-UX	gcc
IBM AIX	gcc
Linux	gcc
Windows 95/98/NT	Borland C++ V4.5
Windows 95/98/NT/ME/XP	Microsoft Visual C++ v5.0, v6.0
Windows 95/98/NT	Cygwin gcc
MacOS-X 10.1 or greater	cc (gcc)

The authors using specific FITS routines of CFITSIO library in a Windows OS environment have developed a simple test program. The main screen of this program is shown in Fig. 5.

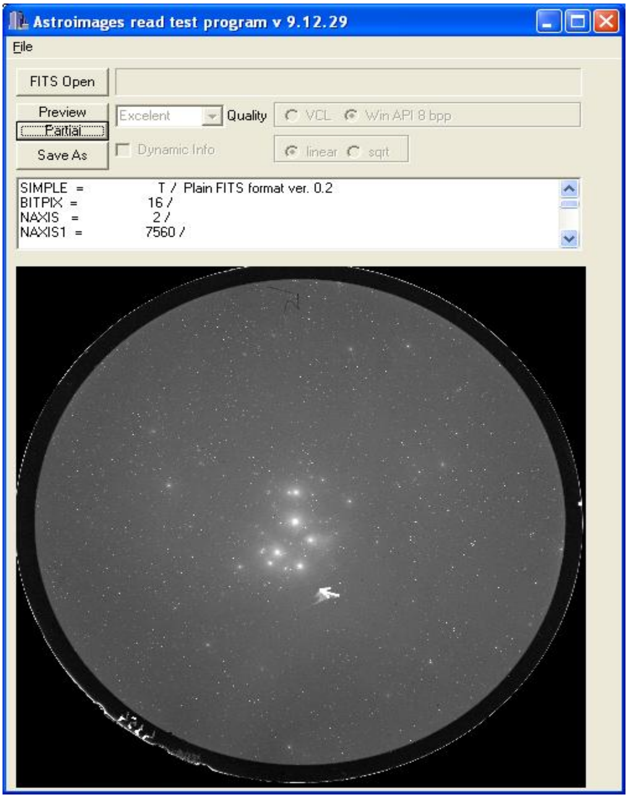

Figure 5. Main screen of the astro-images read test program.

The program opens the specified FITS file, prints out the keywords content of its header in the current HDU (header and data unit) and visualizes a preview of the graphical image content.

5. GEODATABASE

In the Astroinformatics project environment the term “geodatabase” is used to describe main data a subset of observatories and to represent the observatories positions on a world map. In its current version the geodatabase is presented via layers in ESRI shape-file format as follows:

- World layers describe countries, lakes, rivers, sites, etc. that are common geodata widely usable in any GIS (Geographical Information System) software;
- An observatories’ layer is specially designed for the Astroinformatics purposes and takes data from table TBLOBSERVATORIES in our main WFPDB data.

Software code for generating the observatories’ layer has been written using PHP script language.

The necessary graphical web interface to the geodatabase has been developed by the Open Source GPL products: MapServer and Chameleon PHP framework (Kalaglarsky, 2009). We call it AsrtoWeb engine (see Fig. 6).

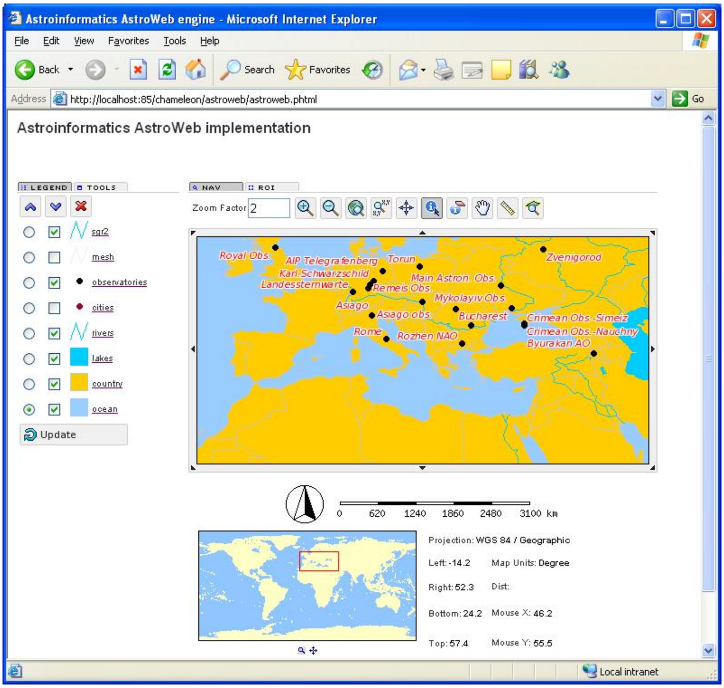

Figure 6. The AstroWeb main screen showing the location of some European observatories.

Fig. 6 demonstrates the Word and Observatories’ layers implemented in a classic web browser, namely the MS Internet Explorer. This web application has full control and nagation capabilities and is also intended to demonstrate graphical and tabular relationship existing in the Astroinformatics project data.

6. APPLICATION SERVER AND ASTRO-IMAGE PROCESSING SOFTWARE UNITS

Astro-image processing software units have to be installed on computer configuration named “Application Server”. These software units will represent pieces of program code, compiled to DLL or EXE format. All information services according astro-image processing will be managed via web interface. The Chameleon PHP framework (Chameleon, 2008) is mainly used in the AstroWeb implementation. Despite of not very known in similar context, the Chameleon PHP occurred very suitable to organize an user friendly web-interface in the frames of a classic HTML browser (with DHTML and AJAX support) and without any special plug-ins. The Chameleon framework itself represents a high customizable program interface based on “widgets”(see Fig. 7).

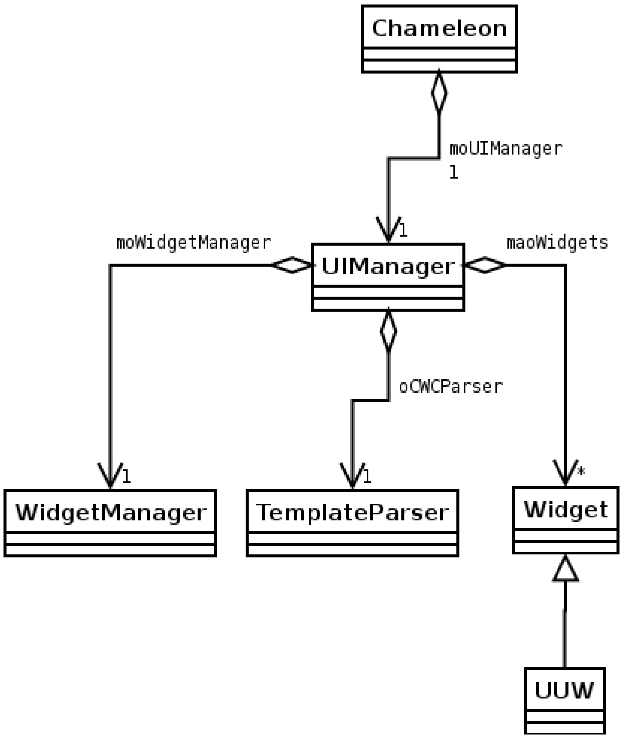

Figure 7. Chameleon framework architecture.

The last block of the diagram in Fig.7 points to a new widget designed according to needed information services. The program developer normally creates new widgets from already existed ones that provide a similar functionality starting from Widget Template and adding new functionalities incrementally. Besides the latter are often interface functionalities, but not necessarily action ones. A very important step is to create the wrapper unit that provides program interface between server-side PHP script implementation and DLLs that support astro-image processing. This task is similar to the one of creation a PHP extended DLL that can be performed using ZEND program interface (Extension writing Part I, 2009). Another Astroinformatics module, namely the wrapper module “AstroWrapper” is under current development. AstroWrapper is intended to link the WFPDB, the Geodatabase and the FITS image content together in a common web-based distributed computer system.

7. CONCLUSIONS

The present paper describes main modules that a R&D astro-image processing environment should generally consist of. The problem solutions here described can be resumed and ordered by their importance to closed future extension as follows:

- Software implementation of CFITSIO open source library for the case of an effective reading and writing of astro-images in FITS format;
- Development of AstroWrapper PHP extension that looks like a bridge between the image-processing executive code and the respective web user interface;
- Development of Chameleon widgets playing as a comfortable software tools for a user friendly web interface;
- Extension of the current AstroWeb implementation for an end-user HTML page to make it a high effective shell to access any information services that “Astroinformatics” project will need;
- Choice and implementation of a suitable communication technology to perform dataflow partnership with optimal security requirements among different workplaces.

References

CFITSIO – A FITS File Subroutine Library, GODDARD SPACE FLIGHT CENTER, http://heasarc.gsfc.nasa.gov/docs/software/fitsio/fitsio.html, consulted June 2009.

Chameleon, A highly customizable and adaptable environment for deploying and managing Web mapping applications, http://www.maptools.org/, consulted March 2008.

Extension Writing Part I: Introduction to PHP and Zend, http://devzone.zend.com/article/1021, consulted April 2009.

Kalaglarsky, D.: 2009, Wide-Field Plate Database - Sofia Search Page, http://draco.skyarchive.org/search/, consulted September 2009.

Kolev, A.: 2009, Astroinformatics AstroWeb implementation, http://demo.valex-edu.net/astroweb.php, consulted June 2009.

Kounchev, O., Tsvetkov, M., Dimov, D., et al.: 2009, Astroinformatics: A Synthesis between Astronomical Imaging and Information & Communication Technologies, Modern Trends in Mathematics and Physics, *Bulg. J. Phys.*, **32**, No.2.

Tsvetkov M. K., Stavrev, K. Y., Tsvetkova, K. P., Mutafov, A. S., Semkov, E. H.: 1997, Wide-Field Plate Database, Institute of Astronomy, Bulgarian Academy of Sciences.