European Southern Observatory

MIDAS Courier

Newsletter of the MIDAS Users' Community

Volume 3, Number 1 July 1993

Now over 200 sites

Editorial

In this ESO-MIDAS *Courier* you will find information about the coming release of ESO-MIDAS, the 93NOV release. Besides descriptions of new applications, you will find important news about system related subjects like the use of shared libraries, the port of ESO-MIDAS to PC's and Graphical User Interfaces (GUI's). Following the ESO Common Conventions, the XHelp has been rewritten while other new GUI's have been added, one of which is the GUI for the display. Other interfaces are now being planned and will follow in the near future. All new GUI's will be based on OSF/MOTIF. In addition to the new application packages developed in-house, you can also read about the new HST context written at the Max Planck Institute for Extra-terrestrial Physics in Garching. The ESO-MIDAS Group gratefully acknowledges this contribution.

As can be inferred from this cover page ESO-MIDAS is now running at more than 200 sites. This number was reached this month: User Agreement number 93A200 was signed by the Astronomical Institute of the Wroclaw University, Poland on July 28. By the time this *Courier* is in print the number of User Agreements already increased to 203. The distribution of sites is as follows: 154 in Europe, 16 in Asia-Pacific, 22 in North America, 9 in Latin America, and 2 elsewhere. Clearly, with such a large user community, efficient support may become difficult. Therefore, we were glad to learn that the vast majority of our users present at this years MIDAS Users Meeting were still satisfied with the support they obtain from the ESO-MIDAS Group.

The number of MIDAS users will probably increase considerably in the near future, now that ESO-MIDAS has successfully been ported to Personal Computers running the Santa Cruz Operating System (SCO-UNIX), or the public domain UNIX Linux (the most popular UNIX system available on the network). A separate article describes the port of MIDAS to Linux.

In this ESO-MIDAS Courier:

General, System and Application News

The MIDAS Group, *European Southern Observatory*

General News

Next Release 93NOV

The next release of MIDAS, release 93NOV, is now in the "alpha" state. That means that our current development version has been frozen in mid July and will be made available for internal testing at ESO with the name 93N08 until mid-September. After this period it will become the "beta" release. Tapes containing this release will be distributed to the beta-test sites.

Reports from beta-test sites are expected to come until the beginning of November after which the official release of MIDAS will be distributed to those sites whose request forms for the 93NOV have been received.

The 93NOV release of MIDAS will be installed and verified on the following machines/systems:

- SUN/SunOS 4.1.n and SUN/Solaris 2.1;
- HP/HPIX;
- SG/IRIX;
- IBM/AIX;
- CONVEX;
- VAX/VMS;
- VAX/Open-VMS;
- PC/Linux;
- PC/SCO.

New "anonymous ftp" account for MIDAS

The anonymous ftp (File Transport Protocol) was first announced in the MIDAS *Courier* Volume 1, Number 2, December 1991. Since then, the ESO-IPG has provided the MIDAS user community with software and information related to MIDAS (e.g. copies in PostScript format of the MIDAS *Courier*), using this ftp account.

This service has proved to be very successful, as indicated by the number of connections received every day. Because of its speed and comfort, the ftp service has already been selected by many MIDAS sites to check and retrieve new releases and patches for MIDAS.

Our Internet name is still the same "ftphost.hq.eso.org" or IP address 134.171.8.4. You can enter "anonymous" or "ftp" as username and give your name or e-mail address as password. Under the directory "/midaspub" you will find all public domain information about MIDAS, including patches and sources for most MIDAS packages. The "core" of MIDAS is, however, not available under this directory. It is restricted to MIDAS sites with a valid User Agreement.

Taking effect the first of September we will upgrade our "ftpd" (File Transport Protocol daemon) using a new release of the software. However, together with this upgrade it will be necessary to change our method of providing access to the restricted MIDAS. In our previous modified version of the "ftpd" daemon we implemented a restricted "/midas" directory that contained the complete compressed-tar sources of MIDAS. It was necessary to login as "ftp" or "anonymous" first, and then to sublogin with the "user midas" command, which required a password, to get access to the "/midas" directory.

With the new "ftpd" daemon the policy remains the same, but the restricted sublogin to "/midas" disappears. A new ftp account called "midas" replaces it. With this new approach the generic "ftp" or "anonymous" account grants you ftp access to public ESO directories like "/midaspub" and others. On the other hand, the ftp "midas" account is restricted to only those activated hosts provided with a valid password. Those sites that have been already registered with the old method will be properly informed about the new changes and they will be provided with new passwords to gain access to the new restricted "midas" account. New sites that want to obtain the new official version of MIDAS 93NOV via ftp, please contact Resy de Ruijsscher to activate your site and to provide you with the corresponding password.

Below you will find two examples. The first one shows how to access through the generic "ftp" or "anonymous" account to public directories at ESO. The second example shows how to get access to the new restricted "midas" account and to retrieve a file. In the examples commands typed by the user are underlined; comments are in italics.

```
% ftp 134.171.8.4  (if ftphost.hq.eso.org doesn't work)
connected to mc3.hq.eso.org.
220 mc3 FTP server (Version 2.1aWU(2) Mon Jul 19 13:43:34 MET DST 1993) ready.
Name (ftphost.hq.eos.org:jdp): anonymous  (or ftp)
331 Guest login ok, send your complete e-mail address as password.
Password: jdp@awislpa.goya.es (use your e-mail address)
230 Bla, bla, bla, bla...
230 Please read the file README
230 Guest login ok, access restrictions apply.
ftp> get README  (get the information file)
200 PORT command successful.
150 Opening ASCII mode data connection for README (479 bytes).
226 Transfer complete.
local: README remote: README
495 bytes received in 0.0032 seconds (1.5e+02 Kbytes/s)
ftp> quit (exit the ftp session)
221 Goodbye.
```

```
% ftp 134.171.8.4  (if ftphost.hq.eso.org doesn't work)
connected to mc3.hq.eso.org.
220 mc3 FTP server (Version 2.1aWU(2) Mon Jul 19 13:43:34 MET DST 1993) ready.
Name (ftphost.hq.eos.org:jdp): midas  (restricted ftp account)
331 Password required for midas:??????
230 This is the restricted midas account (password supplied by ESO-IPG)
230 Please read the file README
230 Guest login ok, access restrictions apply.
ftp> get README  (get the information file)
200 PORT command successful.
150 Opening ASCII mode data connection for README (779 bytes).
226 Transfer complete.
local: README remote: README
495 bytes received in 0.0032 seconds (1.5e+02 Kbytes/s)
ftp> quit  (exit the ftp session)
221 Goodbye.
```

Patches for the 92NOV release

Since the previous issue of the ESO-MIDAS *Courier* no new patch file for the 92NOV version was put on the anonymous ftp account. (`ftphost.hq.eso.org` or IP 134.171.8.4) under the `midaspub` directory. Hence, the patch file P92NOV.01 is still the most recent one. Distribution tapes of ESO-MIDAS contains, besides the latest official release of ESO-MIDAS, those patches that have been released at the time the tape was generated. They are included on the distribution tape as a separate tar-format file. When patches are included in the distribution tape, the sources of the command "patch" are also included as the third tar file.

Patches will not be distributed on tape to sites that have already received the distribution tape, except for very particular cases (e.g. those sites without access to the Internet). Sites that have received and installed the distribution tape are strongly recommended to check for the existence of new patches in our "anonymous ftp" account under Important.

System News

Announcement for the 94NOV release

The MIDAS Group proposes to modify the type definitions of arguments in the C-implementation of the Standard interfaces as of the 94NOV release of MIDAS (November 1994). Elsewhere in this ESO-MIDAS *Courier* you will find more detailed information about this proposal.

MIDAS installed on Solaris 2.1

MIDAS has been installed and verified successfully on a SUN SparcClassic running the new System V operating system from SUN Solaris 2.1.

The installation was done using the public domain GNU C compiler, which is available for Solaris 2.1 on "anonymous ftp" accounts, and the SUN licensed Fortran compiler SC2.0.1. The installation will be also tested using the SUN licensed C compiler.

The only problem with the port of MIDAS to Solaris 2.1 worth mentioning was the misalignment by default of double-precision data types, which is incompatible with the alignment expected in the C code. This problem is solved with the new "-dalign" option in the Fortran compiler, which is included automatically with the new Solaris 2.1 preinstallation.

Shared Libraries

New in 93NOV is the installation of MIDAS with shared libraries, which have the major benefit of the reduction of about 50% in the size of binary files. The following is a list of systems where MIDAS is implemented using shared libraries, and other systems will follow this list in the coming months

- SUN/SunOS 4.1.n;
- Solaris 2.1;
- HP/HPIX;
- PC/Linux.

Application News

Graphical User Interfaces

The version 93NOV of MIDAS will incorporate a set of four MOTIF based interfaces: XHelp for the on-line documentation, XDisplay for display-related commands, XLong for 1D and long-slit spectroscopy and XAlice (see below) or spectral analysis. All these interfaces conform to the ESO GUI Common Conventions which define the Look and Feel of all ESO interfaces in the fields of telescope and instrument control, archives and data analysis. The version 92NOV incorporated many Athena based interfaces (XSpectra, XEchelle, XFIlter, XStella). Some of them (XEchelle, XFilter, XStella) have not been yet ported to MOTIF and compiled versions for Sun and HP will be available under anonymous ftp account on our host "ftphost.hq.eso.org".

New Context XALICE

There is a new interface available for interactive analysis of spectral data. It is called XAlice and can be used from MIDAS by typing `crea/gui alice`. It was developed by Juan Veliz at La Silla. The main features are:

- Flux integration: it is possible to integrate the flux of a spectral feature above or below the continuum. A gaussian fit is performed to determine the wavelength of the feature. It computes parameters like: fwhm, equivalent width, flux, continuum, a.s.o;
- Rebinning: logarithmic, inverse and red-shift rebinning are implemented;
- Filtering: smooth and median filtering is allowed;
- Multi-gaussian fitting: gaussian curves of up to 9 components can be fitted after giving the fwhm, the central X value and the amplitude interactively.

New Context WAVELET

The 93NOV release of MIDAS will contain the new context WAVELET. It is written by Jean-Luc Starck, during his three months visit of the MIDAS Group at ESO. The package can be invoked with the command `SET/CONTEXT WAVELET`. It contains about twenty commands, most of which concern images. The command `TUTORIAL/WAVE` gives an overview of the various wavelet transform algorithms, what kind of data they produce, and how to visualize these data. Several wavelet transform algorithms, and tools to manipulate the wavelet transform have been implemented. More information about the new package is described in a separate article in this *Courier*.

New Context DO

A new context DO (for **D**ata **O**rganizer) written by Michèle Peron is almost completed and will be released for internal use soon. The package enables the user to sort and group his data (e.g. data from one or more observing runs), based on a set of selection criteria, and using many of the facilities of the MIDAS table file system. More information about the package can be found in separate article elsewhere in this *Courier.*

Since the software could not be included in the MIDAS beta-test version, it will not be included in the official 93NOV release. However, the package will be available on the MIDAS anonymous ftp account at soon as the testing has completed.

New Context CCD

The DO context is used in the new CCD context, already announced in the last MIDAS Users' Meeting. As the DO package the CCD software will not be included in the official 93NOV release of MIDAS but will be made available on the anonymous ftp account. More details about the package can be found in this *Courier.*

Contexts SPEC and LONG

The package LONG has been totally refurbished since the 92NOV release. It incorporates 1D and long-slit spectroscopy and presents all functionalities previously spread in three different packages (SPEC, SPECTRA and LONG). A tutorial (command `TUTORIAL/LONG`) describes the commands of the package.

The result of this exercise is that the functions of the context SPEC for reduction of 1D spectroscopy data are now covered by the package LONG. The context SPEC still exists but includes only some low-level commands used by both main spectroscopy packages LONG and ECHELLE.

ESO-MIDAS available on PC/Linux

Carlos Guirao, *European Southern Observatory*

In the previous issue of the MIDAS *Courier* of January 1993 we described the efforts of the MIDAS Group to port MIDAS to one of the announced Unix systems for PCs. In particular, the MIDAS Group decided to use Linux, a public domain Unix system, as the most suitable Unix for the MIDAS community. Linux not only satisfied the requirements to install MIDAS (C and Fortran compilers plus X11) but it is in compliance with ANSI-C and POSIX standards and, in addition, it also includes all the network software necessary to integrate the PC into a LAN. Linux is supported by the Free Software Foundation, and collaborates with Linux porting and providing public domain software (like GNU).

The popularity of Linux has increased enormously in the last months, as can be inferred from the "linux" newsgroup, one of the most active newsgroup currently in the USENET service, and with that the interest of the astronomical community for having MIDAS ported to it. This became apparent during the 5th ESO/ST-ECF Data Analysis Workshop, where the MIDAS Group showed the progress with the port of MIDAS to a PC/486 running Linux SLS 1.0.

Now, the situation has again improved substantially and we are pleased to announce that the complete "core" of MIDAS been successfully implemented and verified on Linux SLS 1.01. Some other MIDAS packages have also been tested by their authors (like WAVELET, PEPSYS, ECHELLE, AND LONG) and a prerelease of our next release of MIDAS 93NOV has been already distributed to several test-sites for a complete check out.

The installation of MIDAS with shared libraries (see under General News), that will reduce the needs of disk space for binary files to less than a half, was recently completed. The Graphic User Interface (GUI) packages for Linux are, however, pending of the delivery of Motif 2.1 and the new release of our X-builder for X11 Release 5 (August-September 1993).

Our next official release of MIDAS 93NOV will be fully supported for Linux and the distribution will be done as usual in QIC-24, QIC-150, Exabyte, DAT/DDS and reel-magtapes in "tar" format, and through our "anonymous ftp" account on ftphost.hq.eso.org (134.171.8.4) in "compressed tar" format, under the restricted "midas" ftp account (password is required).

Contact Resy de Ruijsscher to request a tape or to activate a password to login on the "midas ftp" account.

To ease the installation of MIDAS, the MIDAS Group is considering the distribution of already installed copies of MIDAS for PC/Linux. Eventually, this could be done on CD-ROMs.

Our hardware and software configuration in the next table is only given for information and is not intended to be the unique nor the minimum hardware option to run MIDAS. A "LINUX HARDWARE COMPATIBILITY LIST" is available and updated in the USENET newsgroup `comp.os.linux` (latest version 0.6, dated 01 August 1993). It is also available in the file `midaspub/linux.hw.list` under our anonymous ftp account. MIDAS requests only a 386 or 486 CPU, Linux release SLS 1.01 or higher, a minimum of 16 Mbytes of memory but some more is recommended if big images are going to be processed, and 75 to 150 Mbytes of disk space for MIDAS depending on which packages will be installed

PC configuration:

- i486DX/25
- 20 Mbytes RAM
- Adaptec 1542B SCSI board.
- 1 Gbyte SCSI hard disk.
- Western-Digital 8013 Ethernet board.
- Local Bus S3 Video Card.

Linux configuration:

- Linux SLS 1.01 0.99.pl9-1
- cc: GNU compiler 2.3.3 (included)
- f2c: Fortran 77 to C translator 22 (included)
- X11 R5 (included)
- MIDAS beta-release 93NOV (28 Mbytes).

The next table is a comparison between performance between MIDAS on a PC and on a Sparc station2. Here we should remark that on Linux there is no real Fortran compiler but a Fortran-to-C translator, and access to disk on the Sparc2 is about 5 times faster than on our PC.

- MIDAS installation time (only core):
  PC/Linux: 44 min.
  Sparc2: 30 min.
- Size of binaries (only core of MIDAS) (a):
  PC/Linux: 32 Mbytes
  Sparc2: 41 Mbytes
- cwhetstones (b):
  PC/Linux: 10 MIPS
  Sparc2: 10 MIPS
- MIDAS benchmark (c):
  PC/Linux: 2435 sec.
  Sparc2: 2045 sec.
- Wavelet tutorial (d):
  PC/Linux: 313 sec.
  Sparc2: 405 sec.

(a) SunOS was linked using a MIDAS shared library while Linux was not.

(b) Whetstone benchmark in C. "A Synthetic Benchmark" by H.J. Curnow and B.A. Wichman in Computer Journal, Vol 19, Number 1, February 1976.

(c) filter/median on a square gaussian image of 1000x1000.

(d) MIDAS package 100% C code.

In order to limit our administrational overhead, we will not distribute the ESO-MIDAS PC/Linux version to individuals but only to registered sites. Thus,we will giveMIDAS site managers permission to distribute PC versions of MIDAS to people associated with their institute.

MIDAS 94NOV: Modification of Type Definitions

Preben Grosbøl, *European Southern Observatory*

Introduction

The MIDAS Group herewith announces a chnage of type definitions of arguments in the C-implementation of the Standard interfaces as of the 94NOV release of MIDAS (November 1994).

The modification consists of:

- a change of the arguments of type "long int" to "int" in all SC and TC routines;
- a change of the type of the parameter 'unit' (e.g. in SCKRDC) from type "char **" to "int *".

The first modification is required in order to have MIDAS ported to the new generation of 64 bit CPUs (e.g. the Alpha chip from DEC) while the latter will resolve possible alignment problems.

Technical Background

The major difficulty in porting MIDAS to the OSF/1 system on an Alpha chip, has been the new length of “long int” variables in the C programming language which changed from 32-bits to 64-bits.

The new length of “long” variables although consistent along the MIDAS C code, is incompatible with our previous assumption that an “INTEGER” Fortran variable, when passed as an argument to a C routine, is received as a pointer to a “long”.

In other words, the MIDAS Fortran-to-C, and therefore, the Standard interfaces define their arguments as “long” and pointers to “long”. Although this is correct on a 32-bit machine, it will not work on an Alpha because, still in Fortran, “INTEGER” variables are 32-bits.

Other solutions like a particular Fortran-to-C for Alpha or an option of the Alpha Fortran compiler which assigns 64 bits to INTEGER, have been rejected in favour of the much cleaner solution of modifying the C implementation of the MIDAS Standard interfaces where “long” arguments will be re-defined as “int”.

Nothing will change with respect to the Fortran interfaces of MIDAS, so, users who have written MIDAS application programs in Fortran are not affected and may continue reading just for pleasure (?). Even MIDAS applications written in C will not be affected as long as they are running on 32-bit machines, which are currently all CPUs except DEC’s Alpha chip under OSF/1.

With this message we want to inform the MIDAS community and alert any developers of MIDAS applications in the C language (only they are really concerned...). For more information please contact Preben Grosbøl (`pgrosbol@eso.org`) or Klaus Banse (`kbanse@eso.org`).

The ESO-MIDAS Environment document is under configuration control in order to ensure a high degree of stability. The MIDAS Group judges that this proposal will provide a clean and reliable implementation of MIDAS on the next generation of computers with a minimum impact on the user community. It is therefore important that the issue is discussed in detail by users in order to ensure that a change is implemented only with full consensus in the community.

A message concerning these changes has been sent to all MIDAS site managers in the start of July with a deadline for comments of July 31. Since no objections were received the changes are considered accepted and the modifications will be incorporated in the 94NOV release of MIDAS. A new version of the ESO-MIDAS Environment document MID-SPE-ESO-11000-0001 will then be issued in January 1994.

DO: The New Data Organizer Context

Michèle Peron, *European Southern Observatory*

Before being able to actually reduce and analyse a new set of observations, the observer has to prepare, sort out and arrange the data, that is for instance, make a first quality check, classify data according to a set of rules and associate with each science frame a set of relevant calibration frames.

This task can be cumbersome because of the complexity of the instruments and the large number of data files they produce.

A first version of the Data Organizer has been developed using the existing capabilities of the Table File System.

The Observation Summary Table

The Data Organizer uses as input a list of FITS files or MIDAS images as well as a list of FITS keywords or MIDAS descriptors which are considered to be relevant (e.g. exposure time, telescope setting, instrument mode). Each of these keywords is mapped into one column of a table that is called the Observation Summary Table.

The Observation Summary Table (OST), and the corresponding information for a given input file is stored into one of its rows.

Classification of Images

A natural way to get an overview of the data one has is to try to classify the files into a set of groups. One may want for instance to group the frames according to the exposure type or one may need to put together all the files observed in a given instrument mode. An instruction for grouping Flat Field exposures could be: Select all files which have a descriptor IDENT matching one of the substrings 'FF', 'SKYFL' or 'FLAT' and can be thought as a logical expression understood by the `SELECT/TABLE` command. A set of default rules may be provided for the classification purpose but the possibility should also be given to the user to write his/her own rules because what is written in the observation log file usually doesn't follow any standard conventions, and for instance, the rule mentioned above would not flag as such flat field exposures which have been identified with the string "f.f". Furthermore, writing such a expression can be tedious unless the user is provided with a utility which facilitates the formulation of the query. An interface using the Table editor has been designed. It enables the user to enter interactively constraints on the existing fields of the OST. Relational operators (e.g. >, <, != or =) as well as values, range of values and wild card characters (*) may be used. The given constraints are ANDed and translated into a expression understandable by the `SELECT` command. This expression is stored in the header of the table. In addition, a command exists to edit it interactively.

The classification process tags each frame with a set of character strings provided by the user that identify the image in a unique way and that will be stored into columns of the OST. For instance a scientific exposure observed in the Red Imaging Mode of EMMI using the filter number 505 may be described with the two strings "SCI" and "FR505".

At this stage of the process, it is already possible to make a preliminary statistical analysis of the observing run and this without having to inspect interactively individual files. One can, for instance, plot the sequence number of the images against the date and use different symbols according to the exposure type in order to have an overview of what has been observed. Or one may want to plot the r.m.s. noise of the flat field exposures against their mean pixel value to check the general consistency of the frames and whether there are any outliers.

Association of Images

The association of science frames with suitable calibration frames is achieved by using the same rule generating interface as referred to above even though the rules to be applied are different: One may want for instance to look for all the flat fields which have been taken within a certain time interval of a given science exposure. It only makes sense to associate to a given scientific frame calibration frames observed with the same telescope/ instruments configuration, in other words images might be associated only if they belong to the same classes that have been defined during the association process. This list of classes, for instance optical-path class, detector mode class has to be provided in the command line.

The Association Process creates a MIDAS table which be used by any reduction package. It contains one column for each of the type of exposures (SC, BIAS, DK,WCAL, etc.) and each row contains for the corresponding science image the set of suitable calibration frames.

The Context DO within MIDAS

The Data Organizer may be used by activating the context DO within MIDAS (`set/context DO`). The following commands are available:

- `CREATE/OST`: create an Observation Summary Table (OST) from a set of FITS files or MIDAS images (`.bdf`);
- `CREATE/RULE`: create classification rules for an OST;
- `APPLY/RULE`: apply a given rule to a OST;
- `ASSOCIATE/IMA`: associate suitable calibrations frames to a scientific exposure;
- `GROUP/ROW`: group the rows of a table by the value of one of its column.

Deadline for the next ESO-MIDAS Courier

The next issue of the ESO-MIDAS Courier, Volume 3, Number 2, will be published in January 1994. This issue will include more detailed information about the new 93NOV release as well as on-going projects of the MIDAS Group. Again, we hope to include contributions by MIDAS users or software developers. Please, send your contributions, preferably in computer-readable format, to the MIDAS account or to the editor before November 30th (see back page for mail address).

A New CCD Context in MIDAS

Rein H. Warmels, *European Southern Observatory*

Introduction

Since the last update of the MIDAS CCD package in 1986 a number of new instruments have been installed on the La Silla Telescopes. In addition, new CCD's became available offering large pixel areas and higher quantum efficiency. With these innovations the variety of observing modes has grown and, as an obvious consequence, the amount and the diversity of data taken have dramatically increased. It is clear that the MIDAS CCD reduction software should be able to cope with these improvements and hence requires compatibility with the hardware as it exists now.

Requirements

When designing the basic layout of the CCD software the following basic requirements were kept in mind:

- it should be robust;
- it should be user friendly, easy to use;
- it should offer processing possibilities for a large variety of instruments;
- it should be able to operate both in an automatic mode (to handling large quantities of data) as well as in single command mode (for single image analysis);
- it should offer flexible reduction procedures;
- it should be intelligent.

How intelligent the system is depends on the capabilities of other parts in the data acquisition (ESO Archive) and reduction (MIDAS) systems. In this respect the development of the CCD package took place at the right time: the new MIDAS Data Organizer package now offers the possibility to create a MIDAS table containing the science frames to be reduced and their associated calibration frames.

General Overview

The CCD package provides a set of basic commands that perform the various calibration steps. These are combining of calibration frames, subtraction of the bias level determined from the overscan area or from a separate bias frame, correction for dark current, division by the flat field, correction for illumination, and correction for the fringe pattern. Also, tools are provided for trimming the frames of the unwanted overscan strip, and for correcting the frame for bad pixels intensities.

All operations steps successfully finished are recorded in the descriptor of the (partially) reduced frame. That facility, which includes updating the `HISTORY` descriptor, avoids repetition of reduction sequences, and provides the user with the information what has been done to the data.

By combining the basic reduction steps a complete reduction pipeline procedure is built, that enables the user to do a completely automatic reduction of all science frames. The pipeline procedure is controlled by a set of reduction keywords, that control which of the available reduction steps will be executed. Obviously, in order to get the desired result, these keywords should be set correctly.

Apart from commands that do the actual work, a number of commands will help the user to manage keywords and descriptors and their content/value.

Basically, this means displaying and changing parameters. Also, commands exist to store the current parameter settings and to retrieve these after a session is restarted.

When the context CCD is enabled, a keyword structure is defined, containing all CCD specific keywords required to execute the various reduction steps. The keywords are roughly grouped according to the reduction step. In the reduction process the system needs specific information stored in the frame descriptor. E.g. the standard MIDAS way to store the exposure time is in the descriptor `O_TIME`. However, in case this information is stored somewhere else the system should be able to find it. Therefore, the CCD keyword structure is equipped with a number of keywords that contain the names of important frame descriptors. If these descriptors are absent sensible defaults will be used. Finally, a number of keywords are created to keep general information about the observation and the status of the reduction.

Handling HST Data in MIDAS

M. Almudena Prieto, *Max Planck Institute for Extraterrestrial Physics, Garching bei München*

After a little more of three years of HST operation the HST archive contains at the present time about 28000 “public” observations. Having expired their proprietary period of one year, any of these observations become available to the general astronomical community and the interested user can access and retrieve his/her preferred ones via STARCAT[1]. The availability of such, still increasing, amounts of data requires MIDAS to provide their users with adequate procedures to read and manipulate HST data in the MIDAS environment.

Specific MIDAS procedures for analysing HST data, either spectroscopy or imaging, are now available. These routines are supposed to form part, in a future, of a much general MIDAS context for handling HST data. Currently, they allow the user to convert a set of HST data, extracted from a given instrument and observing mode, in the appropriate scientific format to be manipulated with the usual MIDAS tools. In their present stage, they only operate with the final, science calibrated data, therefore, no attempt for any re-calibration is made. Presently, they are only implemented for their most frequent observing modes of HST. The instruments considered are FOC, WF/PC, FOS and GHRS.

The general structure of the procedures is the following one: read FITS format versions of the HST data into MIDAS, rename the files to their associate HST observing names, and finally, transform the files to an adequate scientific format to be displayed and manipulated in MIDAS (e.g., creation of a WF/PC mosaic out of the four corresponding CCD images). Each instrument has associated its specific MIDAS procedures to perform those tasks.

Interested users can get the package of procedures by anonymous ftp in mc3.hq.eso.org (134.171.8.4). For UNIX users, the package plus a README are in a compressed tar file called: `midaspub/92NOV/unix/hst.tar.Z` For VMS users, the procedures are located in: `midaspub/92NOV/vms/hst/*.prg`. A README is also available in the `hst` subdirectory.

The README file contains a detailed description of what each procedure does and how they must be used. Potential users should follow the steps indicated in order to guarantee a successful application. Questions and comments are welcome; please, send a mail to either the MIDAS group or to the author at `alm@mpe-garching.mpg.de`.

1. STARCAT (Space Telescope ARchive and CATalogue) has been jointly developed by the ESO/Archive, the ST-ECF Archive and the CADC Science Data Archives.

Wavelet: A New MIDAS Context

Jean-Luc Starck, *European Southern Observatory*
Fionn Murtagh, *Space Telescope-European Coordinating Facility*

Introduction

The Fourier transform is a tool widely used for many scientific purposes, but it is well suited only to the study of stationary signals where all frequencies have an infinite coherence time. The Fourier analysis brings only global information which is not sufficient to detect compact patterns. Gabor (1946) introduced a local Fourier analysis, taking into account a sliding window, leading to a time frequency-analysis. This method is only applicable to situations where the coherence time is independent of the frequency. This is the case for instance for singing signals which have their coherence time determined by the geometry of the oral cavity. Morlet introduced the Wavelet Transform in order to have a coherence time proportional to the period.

The continuous wavelet transform of a function *f(x)* is described by Grossmann and Morlet (1984):

$$W(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} \psi^*\left(\frac{x-b}{a}\right) f(x)\, dx$$

where a (>0) is a dilatation parameter and b is a translation parameter. The transform is characterized by three main properties: i) it is a linear transformation, ii) it is covariant under translations, iii) it is covariant under dilations. We get in the Fourier space:

$$\hat{C}(a,\nu) = \sqrt{a}\,\hat{\psi}^*(a\nu)\hat{f}(\nu)$$

The transformation is a set of filterings. When the scale a varies, the filter is only reduced or dilated. The function *f(x)* can be restored from its wavelet transform. The restoration formula is:

$$f(x) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{0}^{\infty} \frac{1}{\sqrt{a}} C(a,b)\, \psi\left(\frac{x-b}{a}\right) \frac{da\,db}{a^2}$$

with

$$C_\psi = \int_0^\infty \frac{|\hat{\psi}(\nu)|^2}{\nu} d\nu < \infty$$

This admissibility condition is generally verified if:

$$|\hat{\psi}(0)| = 0$$

Many 2d extensions of the continuous wavelet transform are possible. Mallat (1989) introduced the multi-resolution analysis and proposed a quick algorithm based on the separability of the variables. This method allows us to have a non-redundant transform, so that the number of pixels in the transform is the same as in the image. This is particularly interesting in image compression. Other algorithms exist which are non redundant (the number of pixels in the transform is larger than the number of pixels in the image): "à trous algorithm", pyramidal algorithms. These algorithms have the advantage that they use isotropic wavelets, which are closer to physical patterns.

Some of the technical applications of the use of the wavelet transform to astronomical imaging derived from this scheme are:

- optimal image addition;
- filtering;
- deconvolution;
- structure detection.

Command Overview

Below follows a brief overview of the commands available in the WAVELET context.

`TRANSF/WAVE` computes the wavelet transform of an image. Eight algorithms are implemented: "a trous" algorithm, Mallat's algorithm, pyramidal algorithms, etc. A file is created which contains the result, and it can be visualized by commands `VISUAL/WAVE`, `VISUAL/CUBE`, `VISUAL/CONT`, `VISUAL/SYNT`, `VISUAL/`

PLAN, and VISUAL/PERS. A typical sequence command would be:

- SET/CONTEXT WAVELET
- RANSF/WAVE NGC2997 WNGC
- VISUAL/WAVE WNGC

Here NGC2997 is the input image, and the result is written in the file WNGC. Default parameters are chosen. The step 3 will create three windows on the screen and load each scale of the wavelet transform in a window. Three files are created by the same command (scale_1.bdf, scale_2.bdf, and scale_3.bdf).

HEADER/WAVE writes on the screen information about the wavelet file (which wavelet transform was used, name and size of the image, etc.) and INFO/WAVE computes statistical informations for all scales (minimum, maximum, standard deviation, etc.).

EXTRACT/WAVE creates an MIDAS image from a scale of the wavelet transform, and INSERT/WAVE replaces a scale by an image. By these two commands, we can modify the wavelet transform.

An image can be reconstructed from its wavelet transform by RECONS/WAVE. The following sequence multiplies the first scale of the wavelet transform by 10, so the high frequencies in the reconstructed image (called NGC_REC) will be enhanced.A typical command sequence would be:

- SET/CONTEXT WAVELET
- RANSF/WAVE NGC2997 WNGC
- EXTRACT/WAVE WNGC s1 1
- COMPUTE/IMA s1 = s1*10
- ENTER/WAVE WNGC s1 1 WNGC_OUT
- RECONS/WAVE WNGC_OUT NGC_REC

Three applications using the wavelet transform have been implemented: Filtering (FILTER/WAVE), deconvolution (LUCY/WAVE, GRAD/WAVE, CIITER/WAVE, DIRECT/WAVE, and comparison (COMPARISON/WAVE, PLOT/SNR, PLOT/COR) of two images.

Two commands concerns one dimensional data: TRAN1D/WAVE which computes 1D wavelet transform and REC1D /WAVE which reconstructs the signal at the full resolution from its wavelet transform.

More information about these commands can be found in the MIDAS Manual Vol 2, release 93NOV.

A Few Applications

After the package was installed is was extensively used for a number of objectives. These objectives were concentrated on image restoration and multi-resolution detection techniques.

Image restoration with simultaneous de-noising.

Thresholding the wavelet-transformed image has been a strategy proposed by a number of authors for noise removal. In a 1992 wavelets conference, Donoho (Stanford) stated: "... shrinking noisy wavelet coefficients via thresholding offers very attractive alternatives to existing methods of recovering signals from noisy data. .... Agreement of diverse theoretical and empirical work is very encouraging and suggests that wavelets will soon have a large impact on how scientists treat noisy data."

An enhanced Richardson-Lucy image restoration or sharpening implementation is available with the command LUCY/WAVE. If no thresholding is carried out in wavelet space, then a classical Richardson-Lucy restoration is obtained. An investigation of the use of this enhanced RL method for de-noising is presented in a paper by Starck and Murtagh ("Image restoration or sharpening with noise suppression using the wavelet transform", submitted). For removing noise, and improving the visual appearance of restored images, the results can be most impressive. In the paper cited, a study is also carried out as to how and where this de-noising approach improves the quality of the basic RL method. Simulated Hubble Space Telescope data is used for this appraisal, and it is found that the quality of faint object detections is significantly improved.

From another perspective, this de-noising strategy additionally solves another problem. This relates to how the user often wishes to treat the background of the image differently from the foreground objects of interest. Greater smoothing in the background may actually be helpful, both in terms of avoiding the creation of artifacts, and also in terms of avoiding unnecessary flux redistribution. But why should one simply view an astronomical image as background plus superimposed objects? Why not recognize that there is often a continuum of superimposed objects? The wavelet transform naturally allows a set of multiresolution transforms of the given image. The de-noising strategy is implemented at these various resolution scales. This contributes in part to the power and effectiveness of wavelet-based image restoration.

Using multiresolution to detect very faint extended objects

The multiresolution views of an image are very valuable, in their own right, for detecting faint structures. For this, we have used the commands `TRANS/WAVE` to create the wavelet-transformed image, using a 6-scale option. Then, we have displayed the resulting resolution scales with the command `VISU/PLAN`. A visual appraisal of these images may lead to the choice of one of them.

The types of faint extended structures examined include cometary comas, and infalling matter in the central regions of galaxies. Images have come from HST Planetary Camera, the NTT SUSI, and the 1.2m at Calar Alto. The wavelet analysis allowed for insightful perspectives on very faint structures in the images. No direct clues were provided in the original images. Examples of this work can be found in Murtagh, Bönhardt, Zeilinger and Starck (“Detection of faint extended structures by multiresolution wavelet analysis”, *Messenger*, August 1993).

A further topic which we hope to check out in the near future is the use of the wavelet transform for detection of spectral lines. The burgeoning literature on the use of the wavelet transform on astronomical data points to its increasingly widespread use. Some of its associated properties, such as its multiresolution treatment of data, make it particularly relevant for astronomical data. All in all, the new wavelet package in MIDAS represents a powerhouse of ideas, many of which are still waiting to be developed and tested.

References

Gabor, D. 1946, Journal of I.E.E. **93**, 429

Grossmann, A., Morlet, J., 1984, SIAM J. Math. Anal, **15,** 723.

Mallat, S., 1989: IEEE Trans. on Pattern Anal. and Math. intel. Vol. **11**, 7

Expeditions in the MIDAS Jungle: Computing

Klaus Banse, *European Southern Observatory*

Finally, we have fought our way through all these details of the data structures in MIDAS and can move on to some of the real stuff now. A data analysis system may have glamorous graphical user interfaces, dazzling graphics and sophisticated typesetting facilities, but that’s no big help if there are not efficient and robust computing tools available to the user. So, this time let’s explore in some depth the commands which support computations involving the different data objects of MIDAS.

Let us begin with the workhorse of the MIDAS commands, `COMPUTE/IMA`[1]. This command works on algebraic expressions written in the standard mathematical way involving MIDAS images and constants. The usual hierarchy of arithmetic operations is applied, to override it use brackets. All calculations are done using single floating point data representation and computation, if an input image has a different data format, the image pixels are converted to floating point format on the fly when the image is read in. The pixels of the output image are stored as real data as well.

Thus, with the command `COMPUTE/IMAGE result = galaxya - 12*(galaxyb-1.1)` we subtract 1.1 from each pixel of image *galaxyb.bdf*, multiply that by 12.0 and subtract all that from the image *galaxya.bdf* to produce the output image *result.bdf*. And the command `COMPUTE/IMAGE result = (galaxya - 12*(galaxyb-1.1))**2`

1. the corresponding command for tables is (you guessed it right!) `COMPUTE/TABLE` but we leave a detailed exploration of that command for a later issue

works as above but additionally takes the square of the data.

The basic mathematical functions, like square root, exponent, log, as well as the main trigonometric functions like sine, cosine, etc. are supported; for a detailed list consult the HELP info of `COMPUTE/IMA`[1].

So, another example would be `COMPUTE/IMAGE result = galaxya - 12*(LOG10(galaxyb)-1.1)`

What happens, if we evaluate a function with a pixel value for which that function is not defined, like taking the log of a negative pixel, or if we divide by an image where some pixels are zero? In such a case the pixel in the output image is set to a predefined *NULL value* and the total number of these *undefined* pixels is reported. The first 2 elements of the real keyword `NULL` are involved here: the *NULL value* is stored in the 2nd element of keyword `NULL` and may be modified by the user (it is set to 0.0 by MIDAS, initially) and the undefined pixel count is stored in `NULL(1)` by the `COMPUTE/IMA` command.

The constants involved in the command may be fixed, like in the example above, or values stored in keywords and descriptors or elements of tables or single pixel values. For example, if you want to divide an image by its maximum value, you would type `COMPUTE/IMAGE image = image/image,{lhcuts(4)}`. For, the real descriptor `LHCUTS` holds the minimum and maximum values of an image in its 3rd and 4th element. Note, that all ‘variables’ in curly brackets “{ }” will be converted to ASCII by the MIDAS monitor before the character input string is fed to the program which executes the command. This conversion is done according to the format indicated via the `SET/FORMAT` command, default values are I4.4 for integer and G15.5 for real and double variables (in FORTRAN notation).

The `COMPUTE/IMA` command takes world coordinates into account (as specified in the descriptors `START` and `STEP` of an image) and works only on those parts of the input images which overlap in world coordinate space. Therefore, all the input images must not necessarily have the same size. The size of the resulting image is set accordingly to the overlapping area of all input frames. Also, the stepsize of all the input images has to match, otherwise an error occurs.

If you just want to do computations with different images on a pixel by pixel basis and ignore their start and step values, use the command `COMPUTE/PIXEL` which has the same syntax and provides the same functionality as `COMPUTE/IMAGE` (but needs input images of same size).

In general, you want to copy all descriptors of the input frame to the result frame. But of which input frame if there are more than one? The default strategy in MIDAS is to copy all (non-standard) descriptors of the *first* input image in the expression[2]. Thus, the descriptors of *galaxya.bdf* would be copied into *result.bdf* in our first example. If, instead, we wanted to copy the descriptors of *galaxyb.bdf* to the result frame, we would have to indicate that via: `COMPUTE/IMAGE result,galaxyb = galaxya - 12*(galaxyb-1.1)`

A few basic mathematical constants can be set via the application procedure `matconst.prg`. The MIDAS command `@a matconst` will create the double precision keyword `MATCONST` and fill its five elements by the values of π, 2π, π/180, e and square root of 2. E.g. if you want to multiply an image with π, enter the command `COMPUTE/IMAGE result = galaxya*{matconst(1)}`.

By the way, `COMPUTE/IMAGE` may also be used as a pocket calculator if only constants are involved by just typing the algebraic expression, e.g. `COMPUTE/IMA log10(123.456)` will print out the resulting value of `2.091512` and also store it into keyword `OUTPUTR(1)`.

The `COMPUTE` commands apply the given calculations to every pixel of the input images. There are many applications, however, where you want to apply different calculations depending upon the value of each individual pixel. Suppose, you don’t want to code your application directly in the FORTRAN or C programming language. You could quite simply do it via a MIDAS procedure by using the `IF` statements and the *immediate* commands involving single pixels[3]. But this would

1. Note, that the trigonometric functions expect their arguments in degrees not radians

2. you can switch off the automatic descriptor copying via the MIDAS command `SET/MIDAS_SYS DSCCOPY=NO`

drive your coffee consumption through the roof while waiting for the results, because the execution of such a procedure is excruciatingly slow except for small one-dimensional spectra

A much better alternative is the MIDAS command `REPLACE/IMAGE` which offers you the possibility to execute those `IF` statements inside a compiled module. As a simple example let us compute an image *result.bdf* which is set to 0.0 for all those pixels of *galaxya.bdf* with a value not greater than 0.0 and equal to *galaxya.bdf* for all other pixels `REPLACE/IMAGE galaxya result <,0.0=0.0` will do exactly that. This means, that for all pixels of the input frame *galaxya.bdf* with a value less than or equal to 0.0 we set the corresponding pixel in *result.bdf* to 0.0; for all other pixels the output pixel will be set to the value of the corresponding pixel of *galaxya.bdf*. The number of pixels falling into the given replacement interval is reported and also stored into keyword `OUTPUTI(15)`.

Instead of setting those pixels of *result.bdf* to the constant 0.0 you might want to set them to the corresponding pixels of image *galaxyb.bdf*. Then, change the command to `REPLACE/IMAGE galaxya result <,0.0=galaxyb`.

In order to replace only those pixels which are equal to 0.0 by the pixels of *galaxyb.bdf* we use `REPLACE/IMAGE galaxya result 0.0,0.0=galaxyb`.

If we want to test the pixel values not in the input frame but in a different frame, e.g. a 0/1 mask *badpix.bdf* indicating bad pixels via 0, we use `REPLACE/IMAGE galaxya result badpix/<,0.0=galaxyb`.

As another example, let us set all the 'bad' pixels to the pixels of *galaxyb.bdf* and all the good ones to the corresponding pixels of *galaxya.bdf* multiplied by 5.3. The command to achieve that is `REPLACE/IMAGE galaxya result badpix/<,0.0=galaxyb,galaxya*5.3`.

And, finally, let us treat the case where we want to set the output frame equal to the input frame *galaxya.bdf* if **all** its pixels are less than the pixels of a second input frame *galaxyb.bdf*, otherwise the output frame should be equal to *galaxyb.bdf*. This cannot be done in a single `REPLACE` command, but a short sequence of MIDAS commands will produce the required output frame:

```
COMPUTE/IMAGE &d = galaxya-galaxyb
REPLACE/IMAGE &d &a 0.0,>=1.0,0.0
COMPUTE/IMAGE &a = {outputi(15)}
REPLACE/IMAGE &a result <,0.0=galaxya,galaxyb
```
[1]

First, we create an intermediate frame *middummd.bdf* containing the difference of the two input frames.Then, we create another dummy frame *middumma.bdf* with the same size as *galaxya.bdf*. The keyword `OUTPUTI(15)` is set to a positive integer if the value of at least one pixel of *middummd.bdf* is greater than or equal to 0.0, i.e. if at least one pixel of *galaxya.bdf* is not less than the corresponding pixel of *galaxyb.bdf*; otherwise `OUTPUTI(15)` is set to 0. In the next command all pixels of that dummy frame are set to `OUTPUTI(15)`, i.e. all pixels are set to 0 if all pixels of *galaxya.bdf* are less than the corresponding pixels of *galaxyb.bdf*, otherwise all pixels are set to a positive integer. Thus, the last command will either set all pixels of *result.bdf* to the values of *galaxya.bdf* or to the ones of *galaxyb.bdf* as required.

Also in the `REPLACE/IMAGE` command all tests and calculations use single floating point data and image pixels are converted to real data on the fly if necessary. But, contrary to the `COMPUTE` commands all frames involved must have the same dimensions (size).

The given examples do not illustrate all the different options available, consult the HELP text of `REPLACE/IMAGE` to obtain a complete list of all possibilities and more examples.

3. see MIDAS Users Guide, Vol. A, Chapter 3 for a detailed discussion of immediate commands

1. this option of `REPLACE/IMA` will only work from release 93NOV on

ESO-MIDAS Request Form

This is a request form for release 93NOV of the ESO-MIDAS[1] system. To obtain a new release, please complete this form and return it to us (see reverse). Material will be shipped only to users with a valid ESO-MIDAS User Agreement. If you still have a tape or tape mailing box from the last release, you MUST return these before we send the new release.

ESO-MIDAS User Agreement No[2]: ______________________________

Technical Support Coordinator: ______________________________

E-Mail: ______________________________

Tape format:

- ○ 1600 bpi VAX/VMS backup
- ○ 6250 bpi VAX/VMS backup
- ○ 1600 bpi TAR format
- ○ 6250 bpi TAR format
- ○ QIC-24 TAR format
- ○ Exabyte 8mm TAR format
- ○ DDS/DAT 4mm TAR format
- ○ DDS/DAT 4mm VAX/VMS backup
- ○ ftp (TAR files only)

Documentation requested:

- ○ ESO-MIDAS User Guide 93NOV
- ○ ESO-MIDAS Environment Version 1.1
- ○ IDI-routines
- ○ AGL Reference Manual Version 3.60

Date: ____________________ Signature: ____________________

---

1. ESO-MIDAS[TM] is copyright protected software developed by the European Southern Observatory for the purpose of Image Processing of Astronomical Data.
2. for new user agreements t.b.d.

The ESO-MIDAS Courier is published twice per year by the ESO-MIDAS Group of the European Southern Observatory.

Contributions as well as suggestions and comments are invited and can be sent to the editor. In particular, authors of ESO-MIDAS application software that would be of general interest for the ESO-MIDAS community, are invited to make this software available.

If you are not on the mailing list and want to receive future issues of the Courier, contact the editor. More updated information about ESO-MIDAS can be obtained from the ESO-MIDAS Bulletin Board, accessible via anonymous ftp.

The ESO-MIDAS™ Courier:

Editor: Rein H. Warmels

Typeset by FrameMaker ™

Published by the European Southern Observatory

Karl-Schwarzschild-Straße 2,

D 85748 Garching bei München,

Federal Republic of Germany

ISSN 1018-3051

The following MIDAS Support services are available for sending suggestions, comments or to obtain help when problems arise.

Telephone:	+49-89-32006456
Telex:	5282822 eso d, attn. ESO-MIDAS
Telefax:	+49-89-3202362, attn. ESO-MIDAS
DECnet/Span:	ESO::MIDAS
Bitnet/EARN:	MIDAS@DGAESO51
UUCP/Eunet:	midas@eso.uucp
Internet:	midas@eso.org
Anonymous ftp:	ftphost.hq.eso.org (134.171.8.4)