GILDAS
An Introduction to the
Grenoble Image and Line Data Analysis System

a GILDAS working group software

March 16, 1999

Version 2.0

The GILDAS working group is a collaborative project of the Observatoire de Grenoble (1,3) and IRAM (2), and comprises: G. Buisson[1], L. Desbats[1], G. Duvert[1], T. Forveille[1], R. Gras[3], S. Guilloteau[1,2], R. Lucas[1,2], and P. Valiron[1].

(1) Laboratoire d'Astrophysique
Observatoire de Grenoble
BP 53 X
414 Rue de la Piscine
F-38402 Saint Martin d'Hères CEDEX

(2) Institut de Radio Astronomie Millimétrique
300 Rue de la Piscine
F-38406 Saint Martin d'Hères

(3) CEPHAG
Observatoire de Grenoble
F-38402 Saint Martin d'Hères CEDEX

Contributions from and invaluable discussions with J. Cernicharo, P. Begou[3], S. Delahaye[3], A. Dutrey[1,2], C. Kahane[1], P. Monger, J.L. Monin[1], R. Neri[2] and all **GILDAS** users are gratefully acknowledged.

Contents

1 Introduction

GILDAS is one of the numerous image processing systems used in Astronomy. **GILDAS** was born sometime ago in the Groupe d'Astrophysique de Grenoble (now LAOG, Laboratoire d'Astronomie de l'Observatoire de Grenoble), and has been adopted as the IRAM standard data reduction package. **GILDAS** is jointly maintained by IRAM & LAOG. **GILDAS** contains many facilities, most of which are oriented towards spectral line mapping, or in fact all kind of 3-dimensional data.

GILDAS has grown on top of two pre-existing programs, **CLASS** and **GreG**, which seemed to please their users. As many other packages, **GILDAS** had to move out of its original VAX-VMS environment towards the growing Unix area. The original software was modified in order to run on VMS and Unix operating systems, with (near) transparent file sharing even between heterogeneous computers. The current implementation even allows remote task submission on a compute server from a workstation. However, because of lack of manpower, VMS is no longer supported. On the other hand, a complete Windows (95 or 98) version is available and supported.

This guide is a general introduction to the **GILDAS** package. It does not describe any utility, but gives cross-reference to other documentations and many usefull hints on how to customize your **GILDAS** environment.

2 Organisation

GILDAS consists of five major parts :

- The documentations, a set of PostScript files. An HTML documentation is also available.
- Dedicated utilities, which are programs using the **SIC** (Sympathetic Interpretor of Commands) user-friendly interface. Each utility normally has its own manual, accessible through the general documentation system.
  - **GreG** an all purpose 1-D and 2-D graphic program
  - **CLASS** (Continuum and Line Analysis Single-dish Software), for single-dish data processing and **CFITS**, a FITS to **CLASS** format translator.
  - **ASTRO**, an astronomical tool, very useful to prepare an observing session, or for amateur astronomy...
  - **CLIC** (Continuum and Line Interferometric Calibration), to calibrate interferometer data from Plateau de Bure.
- Dedicated small programs to perform non interactive time consuming processing : smoothing, transpositions, fitting, etc... These programs will be called "Tasks". They are not intended for interactive use (although you may do it if you are an expert), but require instead a monitor program to activate them.
- Two monitor programs named respectively VECTOR and GRAPHIC,
  - VECTOR is a **SIC** based program which can submit or run in parallel any **GILDAS** task through the use of its `RUN` and `SUBMIT` commands. By using **SIC** vector computing facilities, it also allows direct operations on images.

  - GRAPHIC, a superset of **GreG** and VECTOR, which in addition allows access to some astronomical catalogs.

  GRAPHIC includes the VECTOR program, and hence can run or submit any Task.

- **GFITS** is a FITS translator for exchange of data with the outer world. **CLASS** and **CLIC** have a direct interface with the **GILDAS** *Image* format.

- Additional applications programs, such as MAPPING (a deconvolution tool), which includes GRAPHIC as a subset.

3 Concepts and Useful Hints

3.1 Images

Images and *Tables* are the two most useful concepts in **GILDAS**. Practically all data used by **GILDAS** are stored as *Images* (or *Tables*). An *Image* is a data file containing an array of up to 4 dimensions, and a small but comprehensive header to store the array dimensions, associated coordinates, etc... *Tables* are just 2-d *Images* with only the dimensions indicated in the header.

Images are used everywhere in **GILDAS**. The **SIC** command monitor directly manipulates *Images* through the `DEFINE IMAGE` command. The **GreG** program is able to display *Images* as contour plots with overlaid bitmaps. **CLASS** and **CLIC**, after working with their own data files, produce *Images* for further processing and display. And finally, all *Tasks* use *Images* for input and output.

Because of this importance of *Images*, we recommend the **GILDAS** users to read carefully the corresponding section in the **SIC** manual. Very efficient use of *Images* is possible within **SIC**, but it is also possible to do things a 1000 times more slowly. The ability of **SIC** to perform mathematics on *Images* can solve many problems, and avoid many "on-purpose" programs.

Images can even be initialized easily from external files of many different formats using the `ACCEPT` command of **SIC**, which allows the user to read in a totally flexible way data files to set the content of **SIC** variables.

3.2 Customizing

Although your system manager will provide reasonable defaults, you may wish to customize a few things to your own taste. Customizing the **GILDAS** environment can be done at two levels: Logical Names, and Initialization files for the **SIC** based utilities.

3.2.1 Logical Names

All logical names are placed in a file named `.gag.dico` in your home directory (`$HOME`). The format of the `.gag.dico` file is

```
LOGICAL_NAME1    equivalent_name1
LOGICAL_NAME2:   Equivalent_Name2
```

where the logical names should always be in capital letters, while case matter for equivalent names. The `‘‘:’’` indicates that the logical name is a directory (pathname). Besides logical names which you can use to define special files, or directories, a few peculiar names are used to customize your **GILDAS** environment:

- `GAG_EDIT`
  the name of your preferred text editor (e.g. vi, emacs, vuepad, ved, ...)

- `GAG_HARDCOPY`
  the default type you wish for graphic hardcopies in **GreG**. Major possibilities are

  ```
  PS FAST      "Fast" grey-scale postscript
  PS GREY      "Nice" grey-scale postscript (clipping more accurate)
  PS COLOR      Color postscript
  HPGL          HP-GL language
  ```

  See **GreG** command `DEVICE` for details.

- `GAG_PLOTTER`
  The (queue) name of the plotter you prefer to use by default. Beware that it should be compatible with the description given in `GAG_HARDCOPY`. Funny results can be obtained sending HPGL commands to a PostScript printer.

- `GILDAS_LOCAL:`
  The path for your own **GILDAS** tasks. For advanced users only.

- `GAG_LOG:`
  The path of a directory where you whish to store the log files for all **GILDAS** applications. The default is `$HOME`.

- `GAG_TMP:`
  A path where you whish to store temporary files which may be created by some applications. For optimum performance, use a local disk of the computer.

3.2.2 File Names

Two problems may arise when running **GILDAS** under Unix. The first one is due to the use of the as an option separator in **SIC**, and as a tree/subtree separator in pathnames under Unix. The second is that **SIC** is normally case unsensitive, while Unix is case sensitive. To alleviate these problems, filenames are treated in the following way under **GILDAS**

- By default, automatic conversion to lower case is done

- Logical names (as defined above) can be used to refer to a file name. For example, `GAG_LOG:TOTO.LOG` will be expanded to `/users/me/toto.log` if `/users/me` is the equivalent name of `GAG_LOG:`

- To prevent case translation, the filename should be preceded by an exclamation mark (!). Since the exclamation mark is normally the comment flag in **SIC** commands, the whole filename should be included within double quotes. To illustrate this behaviour, compare these two examples

  ```
    Typed named          Expanded Filename
  "Toto.Log"            toto.log
  "!Toto.Log"           Toto.Log
  "gag_log:Toto.Log"    /users/me/toto.log
  "!gag_log:Toto.Log"   /users/me/Toto.Log
  ```

- As shown above, logical name translation occurs even when an exclamation mark precedes the filename.
- VMS-like names can be typed and will be translated to Unix-like names.

```
  Typed named                    Expanded Filename
GAG_LOG:[SUB.DIRE]TOTO.LOG     /users/me/sub/dire/toto.log
"!gag_log:[SUB.dire]Toto.Dat"  /users/me/SUB/dire/Toto.Dat
```

- DOS-like names can also be typed and will be translated to Unix-like names.

```
  Typed named                    Expanded Filename
\users\me\toto.log             /users/me/toto.log
"!sub\DIRE\Toto.Dat"           sub/DIRE/toto.dat
```

Filename handling is somewhat cumbersome, but becomes very simple when filenames are kept simple and lower case, as often happens.

3.2.3 Initialisation Files

All **SIC** based interactive programs read a initialisation file before starting. This file is a standard command procedure for the corresponding program. For example, for **GreG** it could contain:

```
SIC\SYMBOL CL "CLEAR PLOT"      ! Short cut to clear the plot
SIC\DEFINE DOUBLE SEC/GLOBAL    !
SIC\LET SEC PI|180|3600         ! Second in Radian
SIC\SIC HELP CONTENT            ! Use PostScript file for HELP
GTVL\DEVICE                     ! Prompt for a graphics device
```

The initialisation file is located in your home directory, and its name is ``init.program'' where ``program'' is the default file extension for the application; this is usually just the application name.

4 Getting Started

The normal starting point is then to use the **GreG** graphic program, which will give you a first experience with the **SIC** command monitor. It is recommended to read first the **SIC** and **GreG** *Cookbooks* (15 and 5 pages respectively).

To get really started with image processing, read the **GILDAS** chapters 2 and 3 (*Running Tasks*, and *Displaying Images*) trying to run some of the image processing tasks (8 pages). However, since **GILDAS** is an image processing system, some practice with the GREG2\ **GreG** language will be soon necessary. This practice can be done using the GRAPHIC program, but reading the **GreG** *Manual* will help (12 more pages).

Reference to more specific sections of the **SIC** and **GreG** documentations become only necessary for advanced users, and obviously for programmers. While the user become more and more familiar with **GILDAS** he (or she) will find that constant reference to the documentations is hardly ever necessary. The internal HELP is usually sufficient.