

GDL (GNU Data Language)

Alain Coulais

(LERMA CNRS et Observatoire de Paris)

On behalf of the GDL team:

M. Schellens (head), J. Gales, S. Arabas,
M. Boquier, P. Chanial, P. Messmer, ...

Some students: G. Marshall, N. Galmiche, T. Mermet

Packagers: O. Poplawski (RH), S. Maret (OSX)

Some anonymous contributors and packagers

I don't succeed to be in touch with some former contributors
(see Authors list in package)

Plan

- What is GDL
- Dependences and libraries
- Packaging versus compilation
- Status and know usages
- Examples and performances
- How to contribute
- Online informations
- Summary

What is GDL

Please visit: <http://gnudatalanguage.sourceforge.net/>
GDL (GNU Data Language) is a free IDL (Interactive Data Language) compatible incremental compiler (ie. runs IDL programs). IDL is a registered trademark of ITT Visual Information Solutions.

GDL is code in C++, and use ANTLR

Full syntax compatibility with IDL 6.0

ALL IDL language elements are supported, including:

- * Objects,
- * Pointers,
- * Structs,
- * Arrays,
- * System variables,
- * Common blocks,
- * Assoc variables,
- * All operators,
- * All datatypes

No GUI support (widgets) is implemented so far.

Libraries and dependances

- gcc
- [Readline] (we do have auto-completion in GDL cursor ;-)
- Pplot
- GSL
- [FFTw] (~2 time faster than GSL version ...)
- [ImageMagick]
- [HDF and HDF5]
- [netCDF]
- [python] binding
- And others (libproj4.3 for Earth mapping, ...)

Platforms where GDL is available as a pre-compiled package

- RH, FC and SL (Thanks to [O. Poplawski](#))
- Debian and Ubuntu
- Gentoo
- BSD
- Mac OSX 10.4, 10.5 and 10.6 (Thanks to [S. Maret](#))

BUT (except for RH/FC/SL) precompiled packages are usually **OBSOLETE**.

I strongly recommend to try to compile TGZ, or even better, the CVS versions.

Platforms where GDL can be compiled

I report here only OS with **recent** success

- Most Linux
 - Mandriva
 - RH, FC and SL
 - Debian and Ubuntu <http://doc.ubuntu-fr.org/gdl>
 - Gentoo
 - ...
- Most *BSD
- Mac OSX 10.4, 10.5 and 10.6
 - http://aramis.obspm.fr/~coulais/IDL_et_GDL/GDLonOSX_10.5.6.html
 - Major concern: to compile the good readline lib ;-)

Status

- Syntax OK
- ASTRON lib mostly OK except the last changes (these new functions added in last IDL versions ...)
- MPfit OK
- read/write FITS files OK
- Save/Restore (except on some tricky structures)
- Most numerical functions available (special functions, Voigt, Bessel*, ...)
- Large set of graphical keywords

Please notice it is very easy to add pro/func coded in C++ in the GDL framework (if performance is an issue).

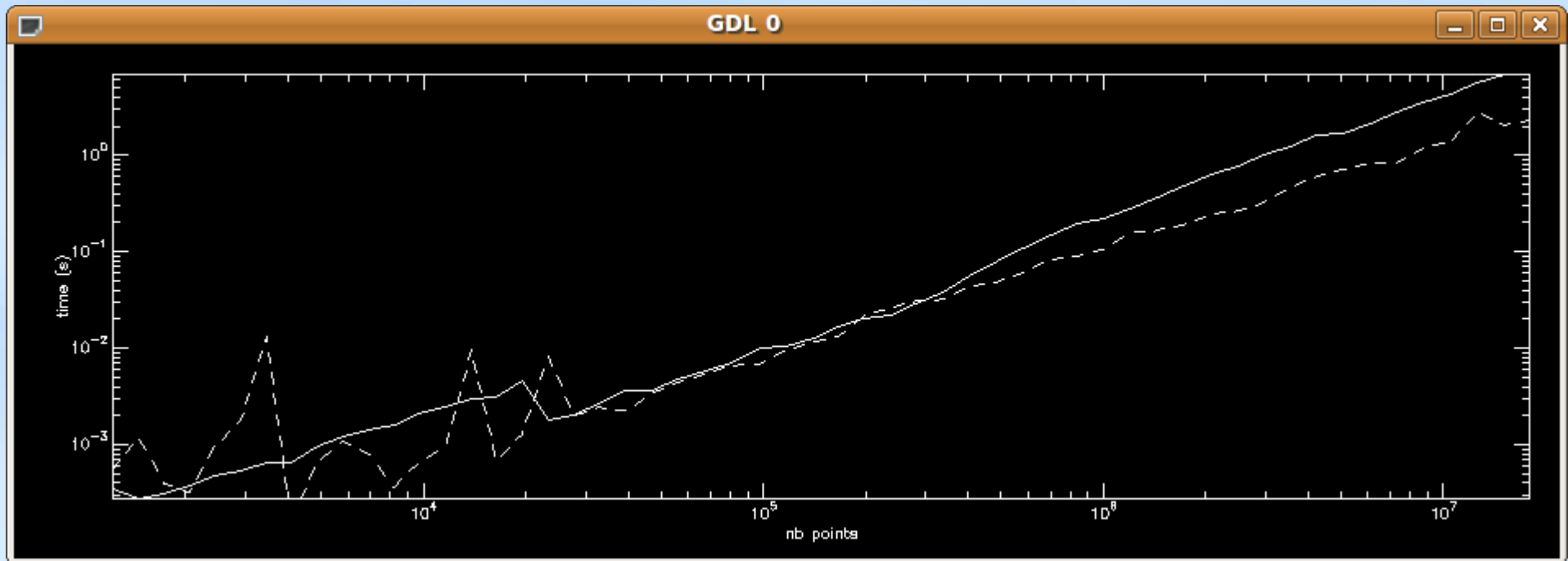
Known Usages

We don't know if GDL is widely used or not, by who and for what but:

- We receive messages worldwide (Europe, USA, Colombia, Israel, India, ...)
- The more the list of pro/func completes the less the messages
- The simpler the compilation the less the messages
- Known usages:
 - Master course at Paris Observatory (~20 students)
 - Course in an university in Poland
 - Refereed papers where computations were done with GDL (e.g. : Koleva & al 2009, Arabas 2008)
 - Computations for PhD Thesis

Examples and performances

- The FFT case
- The TIME_TEST3 case
- Demo (FFT on $2^N \times 3^P \times 5^Q$; dash: GDL, line: IDL, on same machine)



How to contribute

- RTFM. We now have a “quick_start_GDL.sh”.
- We really appreciate code donation ! (under GNU GPL)
- Contributors and Packagers are very welcome !
- Development is clearly driven by end users (you), except for very difficult tasks (“median” code was paid !)
- Don't be shy: we do have people who hesitate to report a bug, missing keyword, problems, ... You can do it anonymously on GDL sourceforge page (bug tracker)
- Don't be impatient: most of us are busy with other tasks
- Please consider the current status is stable enough for some frequent users
- **Please report bugs and regressions !**

Goals for next year

Please remember we consider that the current state of GDL (0.9rc3) represents a good interpreter and a large set of working pro/func.

- Wider set of graphical keywords
- Better Postscript output (or other format(s) for graphical outputs for publications)
- Wider automated testsuite to avoid regressions and better catching bugs
- Better availability of up-to-date pre-compiled packages
- Inline or CLI help

Inline ressources

- Main pages
 - <http://gnudatalanguage.sourceforge.net/>
 - <http://sourceforge.net/projects/gnudatalanguage/>
- Browsing the CVS:
 - <http://gnudatalanguage.cvs.sourceforge.net/viewvc/gnudatalanguage/gdl/>
- Ubuntu
 - <http://doc.ubuntu-fr.org/gdl>
 - <http://packages.ubuntu.com/fr/hardy/interpreters/gnudatalanguage>
- Miscellanea
 - How to Compile:
http://aramis.obspm.fr/~coulais/IDL_et_GDL/memo_GDL.html
 - What is available ?
http://aramis.obspm.fr/~coulais/IDL_et_GDL/Matrice_IDLvsGDL.html

Summary

- GDL is a living project which :
 - does respect IDL syntax
 - does include now a very large set of procedures and functions (in C++)
 - can run most ASTRON codes, MPfit lib, save/restore ...
 - works on x86 and x86_64 (maybe wider, but who care in 2009 ?)
- GDL does run on Linux and OSX.
- On the computation side, GDL is competitive with IDL (functionalities, speed, scalability)
- On the graphical side, GDL is currently not at IDL level (some missing Keywords, limited postscript export) BUT graphic performance with PLplot are close to IDL ones.

Please give a try !