

The GNU Data Language

Alain Coulais

based on Médéric Boquien' talk
last year at RMLL !

GDL? Let's talk about IDL first!

- What?
 - Interactive Data Language
 - Programming language
 - Vectorised, numerical, interactive, imperative, multi-threaded, dynamically typed, interpreted by a virtual machine, large standard library, easy to learn and use
- Who?
 - Proprietary software edited by ITT Vision

GDL? Let's talk about IDL first!

- When?
 - 70s
 - first version ever, for PDP-11 on 1979
 - 80s
 - VAX/VMS (1981) and UNIX (1987)
 - 90s
 - Widgets (1992), Microsoft Windows (1992), Mac OS (1994)
 - 2000s
 - Pointers and objects

GDL? Let's talk about IDL first!

- Why?
 - Data processing
 - Visualisation
- Where
 - NASA, ESA, CEA, CNRS (INSU) etc.
 - Very present in astronomy for historical reasons
 - Remote sensing
 - Medicine

Reasons of the popularity

- Easy syntax
- Very fast learning for someone who already knows Pascal, Fortran, C, etc.
- Rapid development
- Very easy to debug
- “Hands on data”
- Open sources libraries (Astron, MPfit,...)
- Cheap ... at the beginning

Looks nice and mature but ...

- The price!
 - 4500€ for one license
 - 500€ per year for updates
 - Expensive for a lab which can need 10s of licenses!
- It is a proprietary software
 - What if support for my platform is dropped?
 - What if the company goes out of business?
 - Can not see the sources

Let's use another language

- Good solution ... theoretically
 - A large collection of useful scripts exists (ex: the NASA astrolib)
 - Some pipelines are written in IDL
 - Searchers aren't likely to learn yet another language
 - No will
 - No time

Solution: GNU Data Language

- Created in 2003 by Marc Schellens
- Free software (GPL)
- Syntax fully compatible
- Multi platform (Linux/Unix, Mac OS X)
- Reuse of free software components
 - GNU Scientific Library, plplot, FFTW3, Image Magick, Python, etc.)

More and less than IDL

- MORE: Extends IDL
 - GDL modules can be written in Python
 - GDL can be compiled as a Python module
 - You can --rather easily-- include your C++ codes inside GDL
 - At least, you can read the code you use !

More and less than IDL

- LESS: Caveats
 - Widgets not implemented yet
 - The standard library not yet fully implemented
 - 250 routines written so far
 - User documentation is ... minimalist
 - Slightly slower than IDL in average

Status in July 2007 (1/2)

- Rather easy to install on most recent linux distros, but some tricks for Mac OS users
- We have request from Mac OS users
- syntax OK
- Most of the main fonctionnalities are presents
- Many details do not working, or were not extensively tested, expecially keywords for graphic routine

Status in July 2007 (2/2)

- can read and write FITS files (Astro)
- New Users expect that they can run directly big IDL programs in GDL : usually the programs compile, the computation (numerical values) are OK but the plots do not fit the IDL ones.
- Very few people are able to contribute in C++

Help!

- Tell about GDL to IDL users
 - More feedback, more bug reports
 - What missing functions they need
- Write some documentation
- Make packages (.deb, .rpm ...)
- Code
 - Optimisations
 - Widgets
 - Routines (GDL, Python or C++)

Useful links

- <http://www.ittvis.com/idl/>
- <http://gnudatalanguage.sourceforge.net>
- http://en.wikipedia.org/wiki/IDL_programming
- http://aramis.obspm.fr/~coulais/IDL_et_GDL/M

then go to the IDL/GDL section (tutorial)