



pyCGNS.intro/Manual

Release 4.0.1

Marc Poinot

September 09, 2010

CONTENTS

1	Contents	3
1.1	Python module information	3
1.2	Modules	4
1.3	Build and Install	5
2	Tables	7

pyCGNS is a Python package for the [CGNS](#) standard. The package gathers various tools and libraries for end-users and Python application developers.

The package also uses [numpy](#) and *HDF5* you should install before pyCGNS.

CONTENTS

1.1 Python module information

The pyCGNS Python module is a collection of 7 modules around the CGNS standard.

pyCGNS is released under the LGPL license See file COPYING in the root directory of this Python module source tree for license information.

The pyCGNS module now includes (former package names)

VAL - Validator (pyC5) - XML grammar based validation of a CGNStree.py

TRA - Translator (pyCRAB) - Set of translators from/to various formats

MAP - Mapper - Load/save function SIDS/HDF5 from/to CGNStree.py

DAT - DataTracer (pyDAX) - DBMS services for SIDS/HDF5 files

WRA - Wrapper (pyCGNS) - CGNS/MLL and CGNS/ADF python wrapping

PAT - PatterMaker - Full CGNS/SIDS patterns with CGNStree.py

NAV - Navigater (pyS7) - CGNStree.py graphical browser

1.1.1 Release notes

Many changes in this v4 release, you can only use MAP, WRA, PAT and NAV. The other modules, VAL, TRA and DAT are present for archival/development purpose but you should NOT use them.

1.1.2 Changes

- cgnserrors changed to full exceptions at pyCGNS global level

1.1.3 Module dependancies

The pyCGNS modules have dependancies with their brothers. The list below gives you the required modules (or optional) for each of them.

- MAP : None
- PAT : MAP
- WRA : PAT MAP
- NAV : PAT MAP (WRA)

1.1.4 Install notes

NAV depends:

The TkTreectrl module is required. You first need to install tktreectrl (last version tested is tktreectrl-2.2.3) and TkinterTreectrl to map it to Python (last version tested is TkinterTreectrl-0.8)

MAP depends:

The CHLone librarie is required

WRA depends:

CGNS/MLL and CGNS/ADF libraries are required

1.2 Modules

The pyCGNS python package module now includes:

1.2.1 MAPper

MAP (new in v4)

The implementation of the SIDS-to-Python mapping. Provides functions for load/save SIDS/HDF5 from/to CGNStree.py

1.2.2 WRApper

WRA (formerly was pyCGNS)

The Python wrapper for ADF and MLL libraries.

1.2.3 PATternMaker

PAT (formerly was pyCGNS/cgnslib)

A set of functions to create/read/write/modify CGNS/SIDS patterns compilants with SIDS-to-Python.

1.2.4 NAVigater

NAV (formerly was pyS7)

A graphical browser for CGNS trees (ADF/HDF5/Python). Allows tree construction by means of copy/paste and patterns.

1.2.5 DATatracer

DAT (formerly was pyDAX)

A set of command-line tools that use DBMS services for SIDS/HDF5 files.

Warning: This module is experimental and should NOT be used for your applications.

1.2.6 VALidater

VAL (formerly was pyC5)

It is a comand-line tool for CGNS tree verification. It uses an XML grammar.

Warning: This module is experimental and should NOT be used for your applications.

1.2.7 TRAnslater

TRA (formerly was pyCRAB)

It is a set translators from/to various formats, these can be used as functions in your own application or as command-line tools.

Warning: This module is experimental and should NOT be used for your applications.

1.3 Build and Install

The first step of the installation is to make sure you have the required libraries. The mandatory libs are *Python*, *numpy*, *HDF5* and *CHLone*. Then, depending on what you want to build and istall in *pyCGNS*, you may need *libcgns*.

Warning: OUPS ! you mean I don't need *libcgns* for the *CGNS/Python* mapping ?
NO you don't, *CGNS* is a data model (so-called *CGNS/SIDS*) and some mapping definitions of this model (such as *CGNS/HDF* for example). *pyCGNS* uses *CHLone* which is another *CGNS/HDF5* compliant implementation.

1.3.1 Required libraries

- *Python* (starting from v2.4)
- *numpy* (starting v1.1)
- *hdf5* (starting v1.8.5)
- *CHLone* (starting v0.4)

1.3.2 Optional libraries

- *CGNS/MLL (libcgns)* (starting v3.0)

1.3.3 Installation process

Once you have these installed you can proceed with *pyCGNS*. You go into the top directory and you edit the *pyCGNSconfig.py.in* (see [Configuration file contents](#)). You have to set the correct paths and various values such as directory search libs or flags.

Then you run:

```
python setup.py build
```

and then:

```
python setup.py install
```

or:

```
python setup.py install --prefix=/local/tools/installation
```

All the modules of the pyCGNS package are installed and you can now proceed with tutorial examples.

1.3.4 Single module installation

You can ask for a single module installation:

```
python setup.py build --single-module=MAP  
python setup.py install
```

You have to check that this installation doesn't overwrite an existing installation with the other pyCGNS modules.

1.3.5 Configuration file contents

The `pyCGNSconfig.py.in` should work with no modification if you have a standard installation. All you have to declare is the directory in which we can find *Python/numpy/hdf5/CHLone/cgns* libraries.

TABLES

- *genindex*
- *modindex*
- *search*