

Guide to Python/CGNS

Marc Poinot ONERA/DSNA/CS2A

> pyCGNS v4.0.1 introduction June 2009



THE FRENCH AEROSPACE LAB

Copyright © 2006-2010 ONERA, France, All Rights Reserved.

ONERA BP 72, 29 Avenue de la Division Leclerc F-92322 Châtillon Cedex FRANCE

http://www.onera.fr



FOREWORDS

The pyCGNS Python package is a set of tools for the CGNS standard. The tools are all released in the pyCGNS package but you can install and use each tool in a separate way, as far as dependancies for this tool allow you to do so.

CGNS.WRA	Wrapper A Python wrap to so-called midlevel library and adf library. No dependancy to other pyCGNS modules, but it requires the CGNS and HDF5 libraries. Formerly was pyCGNS.MLL and pyCGNS.ADF
CGNS.MAP	Mapper Implements the SIDS-to-python mapping. No dependancy to other pyCGNS modules, but it requires the HDF5 library. Formerly was CGNS.utils.saveAsADF and loadAsADF
CGNS.PAT	PatterMaker A set of functions to create or update complete SIDS sub-trees. No dependancy to other pyCGNS modules. Formerly was pyCGNS.cgnslib
CGNS.NAV	Navigater A Tk based GUI for Python/CGNS trees. Requires CGNS.MAP, CGNS.PAT, TkTreeControl and TkinterTreectrl. Formerly was pyS7
CGNS.VAL	Validater A customizable Python/CGNS tree parser to check tree compliance to SIDS or to user-defined rules. Requires CGNS.MAP Formerly was pyC5
CGNS.DAT	DataTracer A RDBMS-based application to store and manage set of CGNS files. Requires CGNS.MAP and a RDBMS. Formerly was pyDAX
CGNS.APP	Applicater A set of examples, libraries, tools and demos. Requires CGNS.MAP and CGNS.PAT Formerly was pyCRAB



CGNS Standard

The Computational Fluid Dynamics community has a standard for data model definition and disk storage: CGNS (CFD General Notation System), it is an AIAA recommanded practice and it is in the process of being an ISO standard in the frame of ISO/STEP.

△ In this document, we refer to the so-called official CGNS documents and libraries when we use the .org suffix. For example, the CGNS.org documentation means 'the official CGNS documentation or library or tools you can find on the web site www.cgns.org'.

The reference document for CGNS is the so-called SIDS, which stands for *Standard Interface Data Structure*. The reference version is described in the document AIAA-R101A-2006 publicly available on the CGNS web site.

About this document

As we are supporting the CGNS standard, many informations related to the standard are not detailled here. The user should read the CGNS/SIDS to have more complete information about the CGNS data structures.

Each pyCGNS module has its own documentation, their may refer to each other, the introduction document contains common information and an overview of pyCGNS.

The required modules and library versions, used by pyCGNS, can be detected at run-time using the pyCGNSconfig module. These are updated during the installation process.



INTRODUCTION

The pyCGNS python module has started some years ago when ONERA had to use the CGNS library with its Python tools. We developed the MLL/ADF library wrappers. Then, as the use of pyCGNS increases, some companion tools appear. We decided to gather them in a single Python module.

1.1 pyCGNS module

The name of the module is unchanged, each tool has its own 3 letters sub-module: CGNS.WRA, CGNS.MAP ...



2 MODULES

Some modules, such as VAL, DAT or TRA, are .

2.1 Core modules

The top module is MAP.



3 CGNS

We are giving some details about CGNS.

3.1 Open vs Proprietary



4 NUMPY

All pyCGNS modules are using numpy.

4.1 Arrays