GEODIAG

Geoscientifc Diagnostics Open Platform version 1.0.0-rc1

Li Dong

This manual is for GEODIAG, version 1.0.0-rc1.

Table of Contents

1 Introduction

2 Basic usage

2.1 Installation

GEODIAG is version controlled by using `git` and hosted at `GitHub`. It is recommendated to obtain GEODIAG through `git` as

```
git clone https://github.com/lasg-model/geodiag
```

In this way, the local copy can be updated easily by

```
geodiag update
```

After downloading GEODIAG to `<geodiag-root>`, modify the configuration of BASH (e.g. `.bashrc`) as

```
source <geodiag-root>/setup.sh
```

Then login again, you should be able to invoke the command `geodiag`.

2.2 Library usage

2.3 Package usage

Diagnostics is organized as packages. Each package specializes

3 Developing diagnostics package

GEODIAG is designed as an open platform. A **framework** for writing diagnostics package is specified. When a package is written following the standard of this framework, it can be directly plugged into GEODIAG and readily used by other users. This standard does *not* try to put handcuffs on researchers, but eliminate the unnecessary duplication of labor.

BASH script is chosen as the developing language, because almost all the servers provide a uniform BASH shell environment. Ruby and Python are more suitable for constructing the framework, but some servers does not provide them and most researchers are not familiar with them. From now on, `<package-name>` repesents the package name (e.g. `mjo`).

The package is driven through `geodiag` command as shown in [Package usage], page 2. Package developers should provide a `driver.sh` BASH script, which contains `<package-name>_help`, `<package-name>_config` and `<package-name>_run` functions so that the package can be treated as a black box.

Base function	**Description**
`<package-name>_help`	Print help of the package, especially the options.
`<package-name>_config`	Generate a configuration file.
`<package-name>_run`	Run the package.