
HBP Validation Framework - Python Client Documentation

Release 0.6.0

Shailesh Appukuttan

Nov 24, 2020

Contents

1	Quick Overview	1
2	General Info	3
3	Regarding HBP Authentication	5
4	TestLibrary	7
5	ModelCatalog	9
6	Utilities	11

CHAPTER 1

Quick Overview

We discuss here some of the terms used in this documentation.

Model A Model or Model description consists of all the information pertaining to a model excluding details of the source code (i.e. implementation). The model would specify metadata describing the model type and its domain of utility. The source code is specified via the model instance (see below).

Model Instance This defines a particular version of a model by specifying the location of the source code for the model. A model may have multiple versions (model instances) which could vary, for example, in values of their biophysical parameters. Improvements and updates to a model would be considered as different versions (instances) of that particular model.

Test A Test or Test definition consists of all the information pertaining to a test excluding details of the source code (i.e. implementation). The test would specify metadata defining its domain of utility along with other info such as the type of data it handles and the type of score it generates. The source code is specified via the test instance (see below).

Test Instance This defines a particular version of a test by specifying the location of the source code for executing the test. A test may have multiple versions (test instances) which could vary, for example, in the way the simulation is setup or how the score is evaluated. Improvements in the test code would be considered as different versions (instances) of that particular test.

sciunit A Python package that handles testing of models. For more, see: <https://github.com/scidash/sciunit>

Result The outcome of testing a specific model instance with a specific test instance. The result would consist of a score, and possibly additionally output files generated by the test.

More detailed tutorials will be published soon.

For any queries, you can contact:

- Andrew Davison: `andrew.davison@unic.cnrs-gif.fr`
- Shailesh Appukuttan: `shailesh.appukuttan@unic.cnrs-gif.fr`

CHAPTER 2

General Info

- From the above descriptions, it can be identified that *running a particular test for a model* under the validation framework is more accurately described as the *running of a specific test instance for a specific model instance*.
- When running a test, the test metadata and test instance info is typically retrieved from the validation framework. This involves authenticating your HBP login credentials.
- The model being tested can be registered on the *Model Catalog* beforehand, or asked to be registered automatically after the test is complete, just before registering the result on the validation framework.
- Registration of the model and its test results also require authenticating your HBP login credentials.
- It should be noted that an HBP account can be created even by non-HBP users. For more information, please visit: <https://services.humanbrainproject.eu/oidc/account/request>
- Collabs on the HBP Collaboratory can be either public or private. Public Collabs can be accessed by all registered users, whereas private Collabs require the user to be granted permission for access.
- The *Model Catalog* and the *Validation Framework* apps can be added to any Collab. A Collab may have multiple instances of these apps. The apps require to be *configured* by setting the provided filters appropriately before they can be used. These filters restrict the type of data displayed in that particular instance of the app.
- All tests are public, i.e. every test registered on the *Validation Framework* can be seen by all users.
- Models are created inside specific Collab instances of the *Model Catalog* app. The particular app inside which a model was created is termed its *host app*. Similarly, the Collab containing the *host app* is termed the *host Collab*.
- Models can be set as public/private. If public, the model and its associated results are available to all users. If private, it can only be seen by users who have access to the *host Collab*. See table below for summary of access privileges.
- No information can be deleted from the *Model Catalog* and *Validation Framework* apps. In future, an option to *hide* data would be implemented. This would offer users functionality similar to deleting, but with the data being retained in the database back-end.
- Models, model instances, tests and test instances can be edited as long as there are no results associated with them. Results can never be edited!

Regarding HBP Authentication

The Python Client for the Validation Framework attempts to simplify the HBP authentication process. It does this as follows:

On first use, the users have the following options (in order of priority):

1. Setting an environment variable named `HBP_PASS` with your HBP password. On Linux, this can be done as:

```
export HBP_PASS='putyourpasswordhere'
```

Environment variables set like this are only stored temporally. When you exit the running instance of bash by exiting the terminal, they get discarded. To save this permanently, write the above command into `~/.bashrc` or `~/.profile` (you might need to reload these files by, for example, `source ~/.bashrc`)

2. Enter your HBP password when prompted by the Python Client.

Once you do either of the two, the Python Client will save the retrieved token locally on your system. Henceforth, this token would be used for all subsequent requests that require authentication. This approach has been found to significantly speed-up the processing of the requests. If the authentication token expires, or is found invalid, then the user would again be give the above two options.

CHAPTER 4

TestLibrary

CHAPTER 5

ModelCatalog

CHAPTER 6

Utilities
