

# PyXNAT: a Python interface for XNAT

## Submission No:

3899

## Authors:

Y. Schwartz<sup>1</sup>, A Barbot<sup>1</sup>, F Vincent<sup>1</sup>, B Thyreau<sup>1</sup>, G Varoquaux<sup>2</sup>, B Thirion<sup>3</sup>, J Poline<sup>1</sup>

## Institutions:

<sup>1</sup>CEA Neurospin, Paris, France, <sup>2</sup>INSERM Neurospin, Paris, France, <sup>3</sup>INRIA Parietal, Paris, France

## Introduction:

As neuroimaging databases grow in size and complexity, so increases the time researchers spend investigating and managing the data to the expense of data analysis. To automate data management and processing tasks, it is crucial to be able to script the access to a database. We introduce here PyXNAT, a Python module that interacts with The Extensible Neuroimaging Archive Toolkit (XNAT) through native Python calls across multiple operating systems.

## Methods:

XNAT (Marcus, 2004) is an open source software platform designed to manage neuroimaging and associated data. It helps organizing and accessing data that is growing in size and in complexity. The Python language enjoys a growing success in the neuroimaging community (Koetter, 2008), as an alternative or a complement to other analysis tools.

The most common approach to work with a large database such as XNAT is to use a web interface to select a sub-population with a search utility and then download the relevant data locally. Databases may store many variables and it may be challenging to pick out the right data to download from any graphical interface. The File System (FS) subsequently aggregates the transferred data and annotates the data in a consistent and meaningful manner with specific paths and file names. This step is equivalent to converting manually the database in a local FS-based store that lacks advanced search capabilities and has to be synchronized - again manually - with the database.

A communication library can give a direct access to the XNAT server and deal with data management tasks such as keep the local data up to date. Processing scripts accessing a central database are easier to share and to re-use. The vocabulary to describe the data is defined at the database scale which means that it is shared and grasped by a group of users.

## Results:

We have implemented a Python module called PyXNAT on top of a REST API (Representational State Transfer) to communicate with XNAT. It is an open-source project available for download at <http://pypi.python.org/pypi/pyxnat> and documented at <http://packages.python.org/pyxnat>.

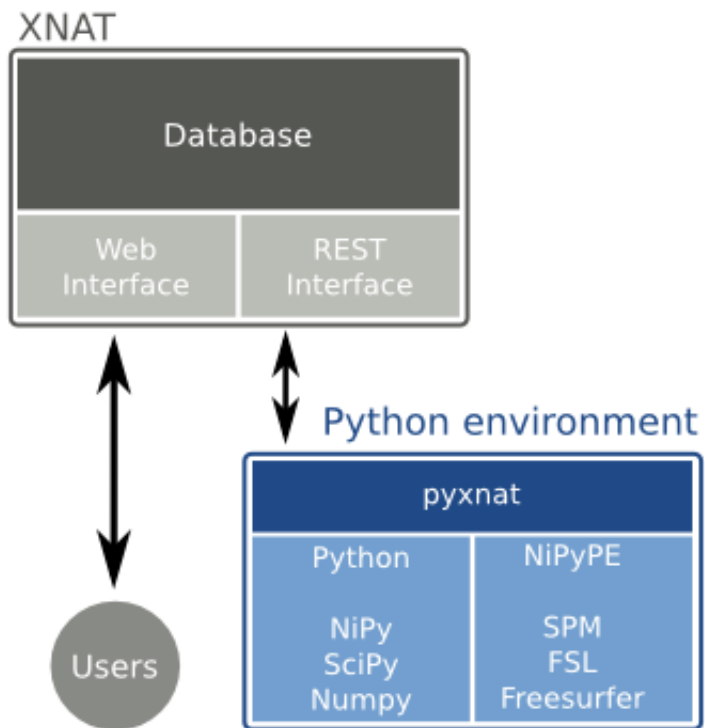
The XNAT REST API identifies uniquely the data with URIs (Uniform Resource Identifier) and uses HTTP for transfer. As a separate feature, the XNAT search engine is also accessible through REST since it can receive an XML document describing a query at a specific URI. Wrapping the REST API in Python makes it possible to unify the two functionalities so that a list of variables and a list of files for a subset of the database are retrievable under consistent semantics. It also introduces new features, such as caching and introspection mechanisms to solve performance issues and help users navigate XNAT.

PyXNAT connects programs to an XNAT server. As an example, NiPyPE is a Python module that interfaces to existing neuroimaging software such as SPM, FSL or FreeSurfer. It is also able to distribute jobs over clusters which makes it very efficient to process large amounts of data. Its data connection method was originally FS-based but it now accesses an XNAT server through PyXNAT. PyXNAT and NiPyPE are being used jointly to run analysis on IMAGEN, which is a European project that aims to study addiction risk factor in 14-year-old adolescents.

## Conclusions:

PyXNAT enables an XNAT access in the Python environment. It can be used both as an interactive command line interface and as a back-end communication library. We see PyXNAT as a major step to help process datasets in XNAT servers. Other projects may use the NiPyPE/PyXNAT combination in the future, such as the International Neuroimaging data-sharing initiative (INDI), a project within the 1000 Functional Connectomes Project.

This work is partly founded by the IMAGEN project from the European Community's Sixth Framework Programme (LSHM-CT-2007-037286). This abstract reflects only the author's views and the Community is not liable for any use that may be made of the information contained therein.



**Figure 1: PyXNAT brings XNAT to the Python environment**

#### Informatics

#### Databasing and Data Sharing

#### Abstract Information

#### References

Ghosh, S., 2010. Nipype: Opensource platform for unified and replicable interaction with existing neuroimaging tools. In 16th Annual Meeting of the Organization for Human Brain Mapping.

Koetter, R., 2008. Python in neuroscience. Front. Neuroinformatics.

Marcus, D., XNAT: A software framework for managing neuroimaging laboratory data. In Proceedings of the 11th annual meeting of the organization for human brain mapping, Toronto, Canada. 12th–16th June. Neuroimage.