Nipype in CBRAIN

Project URL: http://github.com/LORIS

T Glatard

1 Introduction

We aim at the large-scale, automatic sharing of software tools between neuroimaging processing platforms. In this session, we worked on the export of tools from Nipype to CBRAIN.

During the HBM 2015 Hackathon, we worked on the export of software tools from the Nipype workflow engine [1] to the CBRAIN web platform for distributed computing [2]. Our solution allows to export Nipype interfaces to the "Boutiques" description format importable by CBRAIN and pointing to a Docker image, containing the implementation of the interface. The interface and its implementation can be automatically exported from Nipype to CBRAIN (see Fig. 1) and to other platforms supporting Boutiques (e.g. Virtual Imaging Platform and the Pegasus workflow engine).

2 Approach

We developed a tool to export Nipype interfaces to the Boutiques tool description format (step 1. on Fig 1.). nipype2boutiques relies on nipype_cmd (a tool to run Nipype Interfaces as Linux command lines). nipype2boutiques parses the inputs and outputs of a Nipype interface and extracts their name, type, description and position on the nipype_cmd command line. nipype2boutiques then generates a Boutiques descriptor pointing to a docker image where the Nipype interface is available. Once a Nipype interface is exported using nipye2boutiques, it can be imported to CBRAIN.

3 Results

We tested nipype2boutiques on a few Nipype interfaces from the FSL Nipype module. We were able to export 3
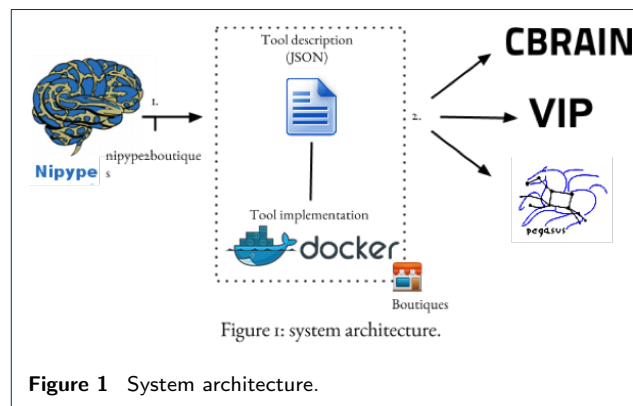

Figure 1 System architecture.

FSL tools automatically from Nipype to CBRAIN, and we verified their correct execution in CBRAIN. Limitations remain on the type of Nipype interface that can be exported by nipype2boutiques: In particular, optional inputs are currently ignored, InputMultiPath are not supported, and output files have to be written in the execution directory of the Nipype Interface.

4 Conclusions

We prototyped a software tool to export Nipype Interfaces as Boutiques descriptors which can be imported by CBRAIN and other platforms. Although the solution is still limited to simple interfaces, we believe that it has the potential to enable fully-automatic tool sharing between Nipype and CBRAIN.

Correspondence: glatard@mcgill.edu.ca

McConnell Brain Imaging Centre, Montréal Neurological Institute, McGill University and Institute of Pscyhology, Montréal, 1 Queen St, 55332, Québec, Canada

Full list of author information is available at the end of the article

Competing interests
None

Author's contributions
TG wrote the software and the report.

References

1. K., G.: Nipype: a flexible, lightweight and extensible neuroimaging data processing framework in python. Frontiers in neuroinformatics **5** (2011)
2. Sherif, T., Rioux, P., Rousseau, M.-E., Kassis, N., Beck, N., Adalat, R., Das, S., Glatard, T., Evans, A.C.: Cbrain: a web-based, distributed computing platform for collaborative neuroimaging research. Frontiers in neuroinformatics **8** (2014)