Summary of the work done during the BrainHack

Dipankar Bachar , David Meunier & Aissam Rahmani

Our project for the BrainHack Marseille was: **Integration of DAX pipeline with XNAT.** Even though we could not achieve all our goals for this BrainHack project, we did advance quite well in this project.

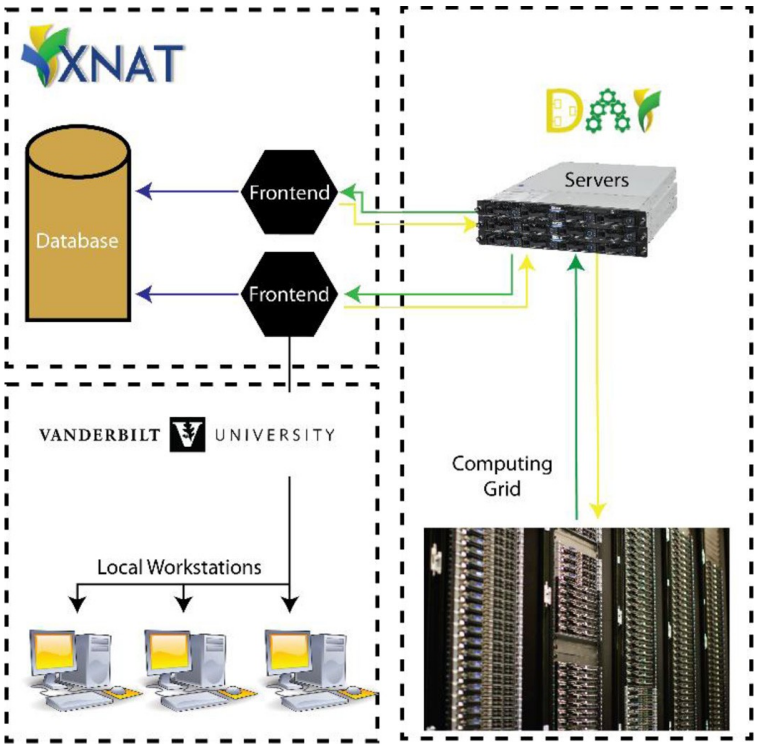

Image 1 : Integration of Dax pipeline with XNAT

During our BrainHacking session we did the following :

1) We prepared an informative and technical presentation that describes DAX.

2) We have re-examined the DAX installation steps and its configuration with XNAT.

3) We did BrainStorming on the various files necessary for DAX such as Spiders, Modules, Processors, SettingsFIle etc.

4) We inspected the docs provided by the DAX development team.

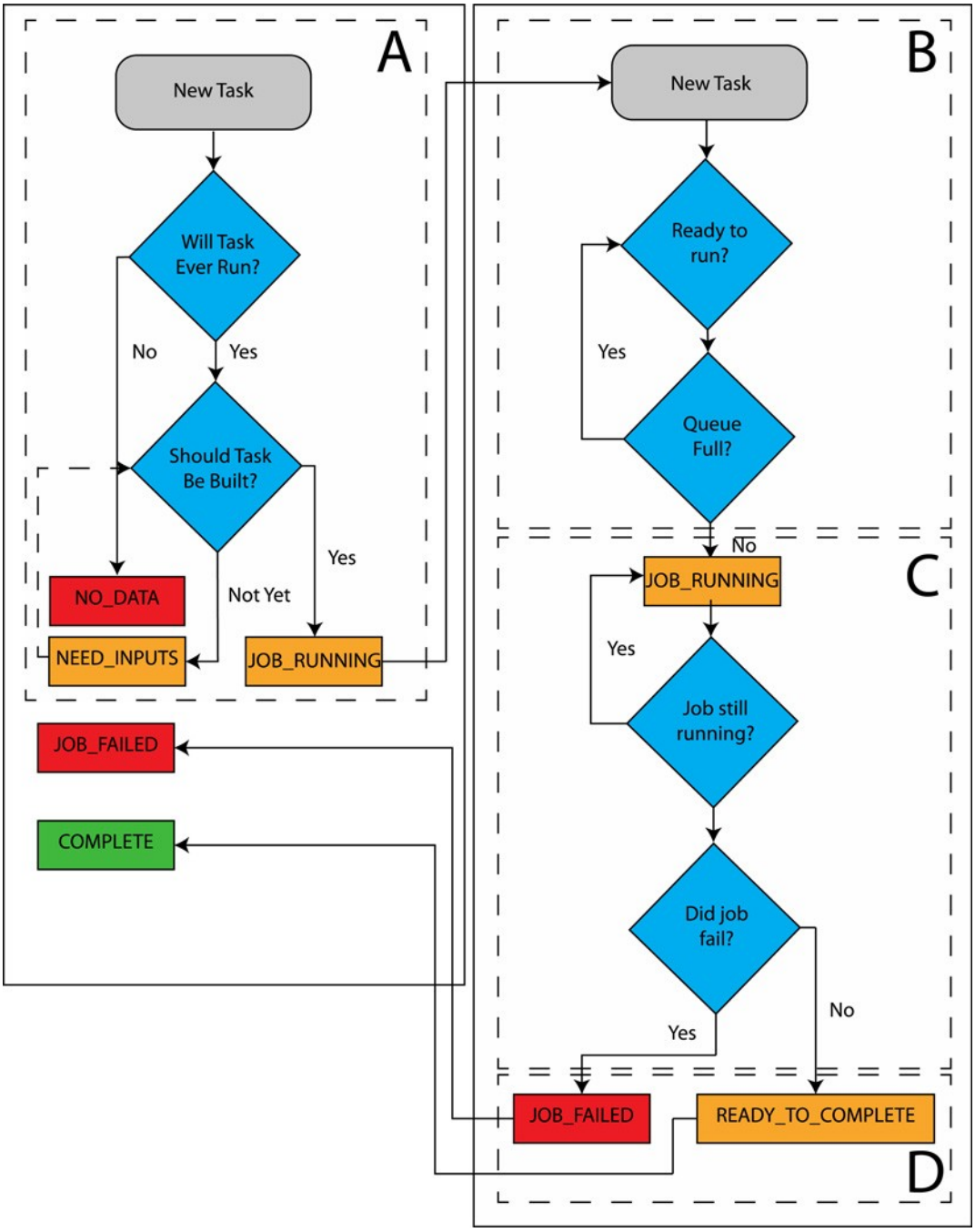

Image 2 : Working of a DAX pipeline

5) We tried to write the Pipeline for DAX. To do so, we go through the documentation provided by the DAX development team. We found that the Documentation of the latest version (DAX 1.41) does not contain enough information to develop these pipelines. On the other hand, the Documentations for older versions (More then 3 years old) contain dedicated instructions for this, but did not work with the latest versions.

6) We managed to understand how to structure and to write a YAML script for DAX processor.

7) We performed a live demo, in order to check DAX & XNAT connection. With execution/launching of all commands using DAX Commandes line tools.

We tested with the two projects : **DrugAddict** and **Ordre.** Here are same example of tests :

Xnatquery -p Ordre : to see the hierarchy of data on Ordre projet

```
------------------------------------------------------------------
Arguments:
  project          -> Ordre
------------------------------------------------------------------

==================================================================
INFO: connection to xnat <http://10.164.4.26:8080/test>:
==================================================================
Project: Ordre
Project: Ordre
  + Subject: 01
  + Subject: 02
  + Subject: 03
  + Subject: 04
  + Subject: 05
======================== Xnatquery DONE ==========================
```

Xnatinfo DrugAddict: to get fast statistics informations on DrugAddict project

```
#     Check the help for examples by running --help             #
#################################################################

------------------------------------------------------------------
Arguments:
  project          -> DrugAddict
------------------------------------------------------------------

==================================================================
INFO: connection to xnat <http://10.164.4.26:8080/test>:
INFO: query through project DrugAddict ...

Information for project DrugAddict on Xnat :
Date: 2020-12-04 13:07:36.927916
================================================================================

Project Info:
--------------------------------------------------------------------------------
Count:
----------------------------------------
Subjects             : 15
Sessions             : 15
Scans                : 313
Assessors/Processes : 0
----------------------------------------

Scan info :
--------------------------------------------------------------------------------
  Scan type                                                                | Count
  -------------------------------------------------------------------------------
  AAHead_Scout_64ch-head-coil                                              | 16
  AAHead_Scout_64ch-head-coil_MPR_cor                                      | 15
  AAHead_Scout_64ch-head-coil_MPR_sag                                      | 15
  AAHead_Scout_64ch-head-coil_MPR_tra                                      | 16
  Fieldmap1_topup_AP                                                       | 2
  Fieldmap1_topup_PA                                                       | 2
  Fieldmap2_topup_AP                                                       | 2
  Fieldmap2_topup_PA                                                       | 2
  Fieldmap_topup_AP                                                        | 28
  Fieldmap_topup_PA                                                        | 28
  T1w                                                                      | 18
  T2w                                                                      | 16
  TEST_1min15                                                              | 13
  TEST_1min15_SBRef                                                        | 13
  task-drugaddict_run1_bold                                                | 15
  task-drugaddict_run1_bold_SBRef                                          | 16
  task-drugaddict_run1_bold_split_1                                        | 1
  task-drugaddict_run2_bold                                                | 16
  task-drugaddict_run2_bold_SBRef                                          | 17
  task-drugaddict_run3_bold                                                | 15
  task-drugaddict_run3_bold_SBRef                                          | 15
  task-drugaddict_run4_bold                                                | 15
  task-drugaddict_run4_bold_SBRef                                          | 15
  task-drugaddict_run5_bold                                                | 1
  task-drugaddict_run5_bold_SBRef                                          | 1
  -------------------------------------------------------------------------------
```

8) Also, we tested a module named **Module_dcm2nii.py** in **DAX** in order to convert our test **xnat data** from **dicom** to **nifti.** As we encountered errors, we did some debugging tasks with this script **Module_dcm2nii.py**. But, due to the time limit of our BrainHack event, we had to stop. We think that these bugs are due to the different version of the python between **DAX**, Module_dcm2nii.py and also may be uncompatible version of **dcm2nii**.

$ dax test –file Module_dcm2nii.py -p Ordre -s O1_m1

Traceback (most recent call last): File "/dax/bin/dax", line 241, in <module> dax_tools.testing(args.test_file, args.project, args.sessions, File "/home/rahamania/miniconda3/envs/dax/lib/python3.8/site-packages/dax/dax_tools_utils.py", line 517, in testing test_obj = load_test(test_file) File "/home/rahamania/miniconda3/envs/dax/lib/python3.8/site-packages/dax/dax_tools_utils.py", line 2186, in load_test return eval('test.{}()'.format(os.path.basename(filepath)[:-3])) File "<string>", line 1, in <module> File "Module_dcm2nii.py", line 35, in **init** self.dcm2nii_exe = check_executable(dcm2nii_exe) File "Module_dcm2nii.py", line 330, in check_executable LOGGER.debug('%s version: %s' % (name, nve_version.strip().split('\n')[0])) TypeError: a bytes-like object is required, not 'str'

Finally, In addition to the technical/computer side. The BrainHack Marseille was an opportunity, which allowed us to learn new aspects on all the projects studied, as well as an acquisition/development of personal skills in our field of work. And finally an enrichment of knowledge.

Links:

DAX : https://dax.readthedocs.io/en/latest/index.html
https://github.com/VUIIS/dax
Xnat :
https://www.xnat.org/

Reference :

DAX - The Next Generation: Towards One Million Processes on Commodity Hardware
Stephen M Damon 1 , Brian D Boyd 2 , Andrew J Plassard 1 , Warren Taylor 2 , Bennett A Landman 1 2