

Analysis with the CLAS12TOOL/ROOT Package in Docker

CLAS Collaboration Meeting March 2019

Adam Thornton

University of Glasgow

08/03/2019

Supervisors: Dave Ireland and Derek Glazier

Outline

- 1 Clas12Tool Docker Tutorial
 - What is Clas12Tool
 - Running Clas12Tool in Docker
 - Starting a Jupyter Notebook
 - Running the Example Notebooks

- 2 Summary

Getting Started

Link to dockerhub repository with instructions:

`https://hub.docker.com/r/ademus4/root-jupyter-haspect`

Data needed for plotting:

`/work/clas12/jnp/clas_004152.recon.hipo` (Jlab work disk)

What is Clas12Tool?

<https://github.com/dglazier/Clas12Tool>

- Collection of analysis tools written in C++ (Root)
- Reading HIPO files directly
- Simple histogram plotting
- Access to all HIPO banks

```
*****
*                               : ----- *
*                               : HIPO 4.0 I/O Library *
*                               : Jefferson National Lab *
*                               : Date: 01/24/2019 *
*                               : ----- *
*****
```

See Derek's presentation for details:

<https://www.jlab.org/indico/event/303/session/3/contribution/23/material/slides/>

Motivation for using Docker



- Entire software installation packaged in container (like a VM)
- Version control of entire set-up
- Portability
- Fast set-up, instant working environment
- Scaling (OSG, local cluster)
- Build containers on containers

Running Clas12Tool Docker

Pull the image from Dockerhub:

Example

```
docker pull ademus4/root-jupyter-haspect:hsfit
```

This will take a minute or so to download. Once finished, it should appear in the list of images on the machine:

Example

```
$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
ademus4/root-jupyter-haspect	hsfit	9ff04f1d1898	2 hours ago	3GB

Running Clas12Tool Docker

The following will run the Docker instance, change the path of `/myfiles/data/` to where you have stored HIPO file:

Example

```
#!/usr/bin/env bash
docker run -it --rm \
  -v /myfiles/data/:/local_work/ \
  -w /work/Clas12Tool/RunRoot/jupy/ \
  -p 8888:8888 \
  --user $(id -u):$(id -g) \
  ademus4/root-jupyter-haspect:hsfit \
  bash
```

The terminal should now show:

Example

```
bash-4.2$
```

Starting a Jupyter Notebook

Within the interactive Docker session, start a Jupyter notebook:

Example

```
jupyter notebook
```

```
bash-4.2$ jupyter notebook
[I 22:10:22.985 NotebookApp] Writing notebook server cookie secret to /work/.local/share/jupyter/runtime/notebook_cookie_secret
[I 22:10:23.270 NotebookApp] Serving notebooks from local directory: /work/Clas12Tool/RunRoot
[I 22:10:23.271 NotebookApp] The Jupyter Notebook is running at:
[I 22:10:23.271 NotebookApp] http://(bb2d990130e6 or 127.0.0.1):8888/?token=22527bfa191eed1951039a1d240c2d6ee90dc4d1dbddb940
[I 22:10:23.271 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 22:10:23.275 NotebookApp]

To access the notebook, open this file in a browser:
    file:///work/.local/share/jupyter/runtime/nbserver-31-open.html
Or copy and paste one of these URLs:
    http://(bb2d990130e6 or 127.0.0.1):8888/?token=22527bfa191eed1951039a1d240c2d6ee90dc4d1dbddb940
xdg-open: no method available for opening 'file:///work/.local/share/jupyter/runtime/nbserver-31-open.html'
```

The notebook will then be available from your browser at the following*:

```
http://0.0.0.0:8888/
```

* Need to copy token from terminal to browser

Starting a Jupyter Notebook

Paste token in browser



Password or token:

Log in

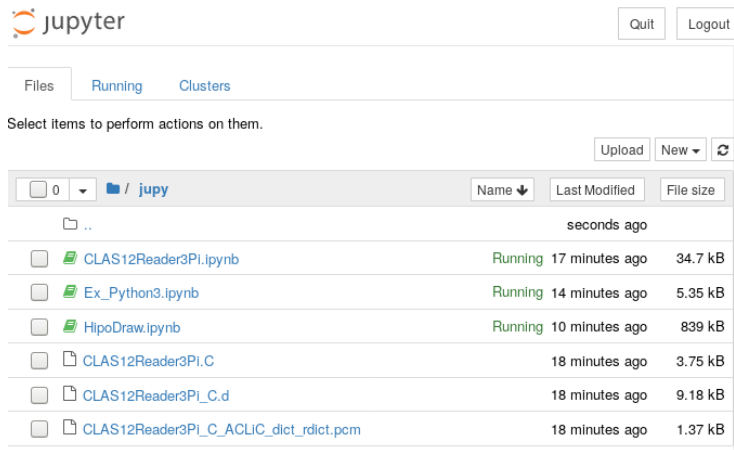
Token authentication is enabled

If no password has been configured, you need to open the notebook server with its login token in the URL, or paste it above. This requirement will be lifted if you [enable a password](#).

The command:

Running the Examples

The working directory contains a number of example notebooks



The screenshot shows the JupyterLab interface. At the top left is the Jupyter logo. On the top right are 'Quit' and 'Logout' buttons. Below the logo are tabs for 'Files', 'Running', and 'Clusters', with 'Running' selected. A message says 'Select items to perform actions on them.' To the right of this message are 'Upload', 'New', and a refresh icon. Below this is a file browser showing the current directory as '/ jupy'. The browser contains a table of files and folders:

<input type="checkbox"/>	0	Name ↓	Last Modified	File size
<input type="checkbox"/>		..	seconds ago	
<input type="checkbox"/>		CLAS12Reader3PI.ipynb	Running 17 minutes ago	34.7 kB
<input type="checkbox"/>		Ex_Python3.ipynb	Running 14 minutes ago	5.35 kB
<input type="checkbox"/>		HipoDraw.ipynb	Running 10 minutes ago	839 kB
<input type="checkbox"/>		CLAS12Reader3PI.C	18 minutes ago	3.75 kB
<input type="checkbox"/>		CLAS12Reader3PI_C.d	18 minutes ago	9.18 kB
<input type="checkbox"/>		CLAS12Reader3PI_C_ACLIC_dict_rdict.pcm	18 minutes ago	1.37 kB

Ex_Python3.ipynb is just a demo for using Python ROOT API, can be ignored for now

Running the Examples

CLAS12Reader3Pi (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted | ROOT C++

General loop over events and particles

```
In [ ]: gROOT->ProcessLine(".x $CLAS12TOOL/RunRoot/hiporoot/LoadHipoROOT.C");
```

Now we can just use the notebook as an editor and change our script if we like.

Open the ROOT script CLAS12Reader3Pi.C

```
In [ ]: %%file CLAS12Reader3Pi.C
#include <cstdlib>
#include <iostream>
#include <chrono>
#include <TFile.h>
#include <TTree.h>
#include <TApplication.h>
#include <TROOT.h>
#include <TDatabasePDG.h>
#include <TLorentzVector.h>
#include <TH1.h>
```

Running the Examples

CLAS12Reader3Pi - Mozilla Firefox

CLAS12Reader3Pi

0.0.0.0:8888/notebooks/jupyter/CLAS12Reader3Pi.ipynb

jupyter CLAS12Reader3Pi Last Checkpoint: 3 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted ROOT C++

```
can->Draw();
}
```

Created file '/work/Clas12Tool/RunRoot/jupyter/CLAS12Reader3Pi.C'.

In [3]: %jsroot

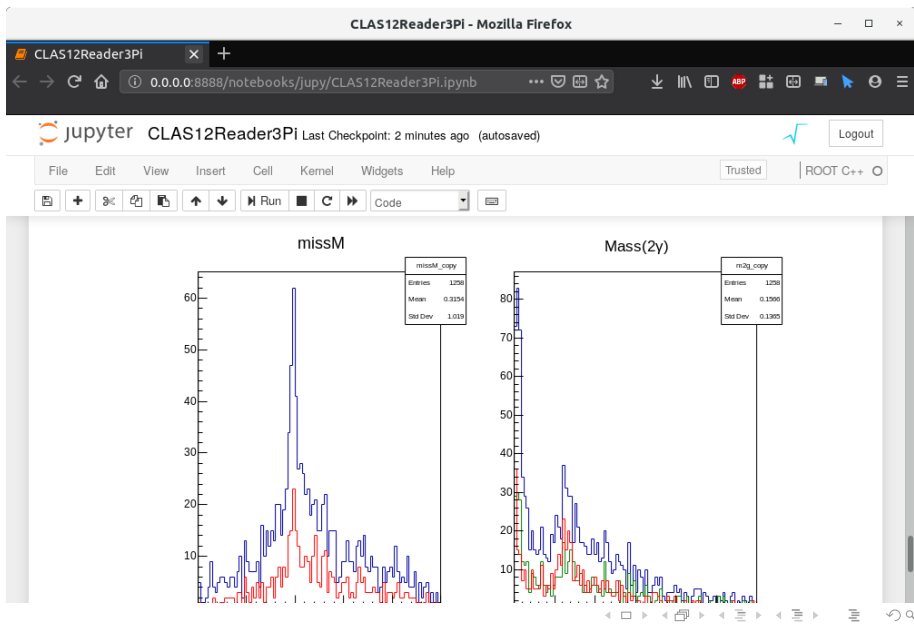
In [4]: gROOT->ProcessLine(".x CLAS12Reader3Pi.C+(\\"/local_work/jlab/data/clas_004152.recon.hipo\\");

Info in <TUnixSystem::ACliC>: creating shared library /work/Clas12Tool/RunRoot/jupyter/./CLAS12Reader3Pi_C.so

```
*****
*          >=<          : ----- *
*  ,---'  '---'  : HIPO 4.0 I/O Library *
*  ( )  '  , _'  : Jefferson National Lab *
*  Xx'xX          : Date: 01/24/2019 *
*****
```

****>>>> compiled with c++11 support.
clas12reader::clas12reader reading /local_work/jlab/data/clas_004152.recon.hipo

Running the Examples



Summary

Follow instructions on the Dockerhub repository below

Links to the various repositories:

`https://hub.docker.com/r/ademus4/root-jupyter-haspect`

`https://github.com/dglazier/Clas12Tool`

`https://github.com/ademus4/Clas12Tool-docker`

Back-up

Docker Run Parameters

`-it` (pseudo-TTY connected to the containers stdin, keeping the session interactive from the terminal, as opposed to detached mode where you run commands separately within the running instance)

`--rm` (removes the container upon exit, keeping things tidy)

`-v` (mounting volumes to be shared between the host machine and docker container, so you can access things from your machine when running the container, and save the work you do)

`-w` (setting the working directory inside the container)

`-p` (mapping a port from inside the container to outside)

`--user` (setting the user, since the default is to run as root, changing files on the local machine will then be unreadable without sudo, and a number of other headaches this can cause, important!!)