

# Hall C Analysis Software Containers

**Anil Panta**

Jan 19, 2024

 Jefferson Lab

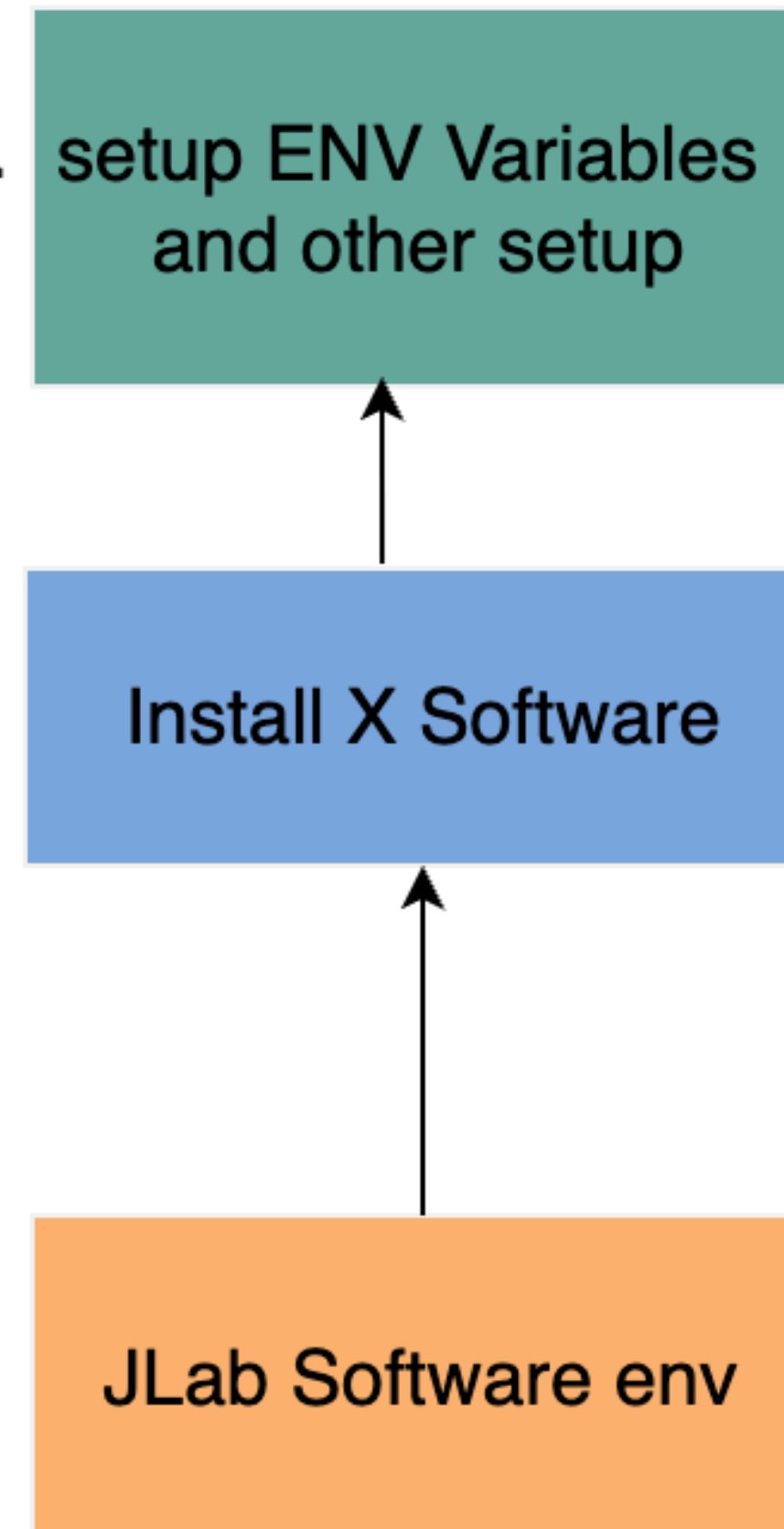
# Software setup from USER POV

---

- > Have to have proper documentation.
- > There is room for errors.

- > Compile the software
- > Select proper release.
- > May run into Issues

- > /site/12gev\_phys \$JLAB\_VERSION
- > Sets root, gcc, evio



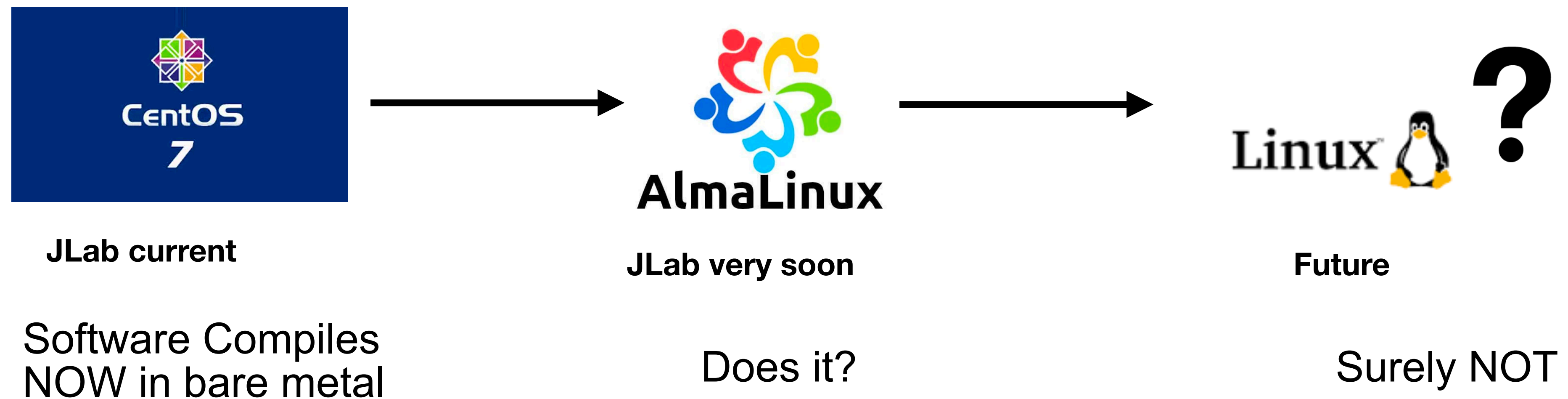
# Software Preservation POV

---

- Current software distribution path
  - Software are installed on physical hardware with no virtualization.
  - All dependencies are installed one by one.
  - Each release and corresponding dependency's release.
    - Should be properly mapped.
    - Results in careful evaluation
- Can we keep on doing this in future too ?
- Can a user in future would be able to use the software?
- Is the software portable to other sites/systems ?

# Challenges

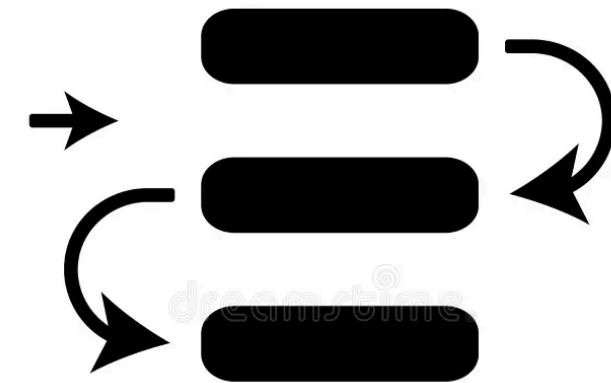
---



NO Guarantee that the same host os will remain in future and software will work

# Challenges

---

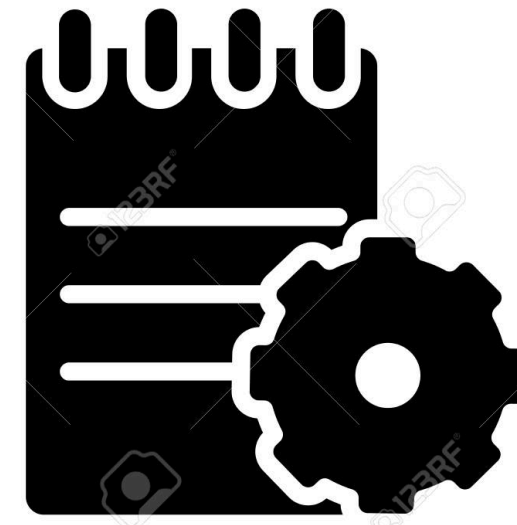


DEPENDENCIES

Dependency NO longer maintained,  
NO backward compatibility changes.

# Challenges

---



COMPILER

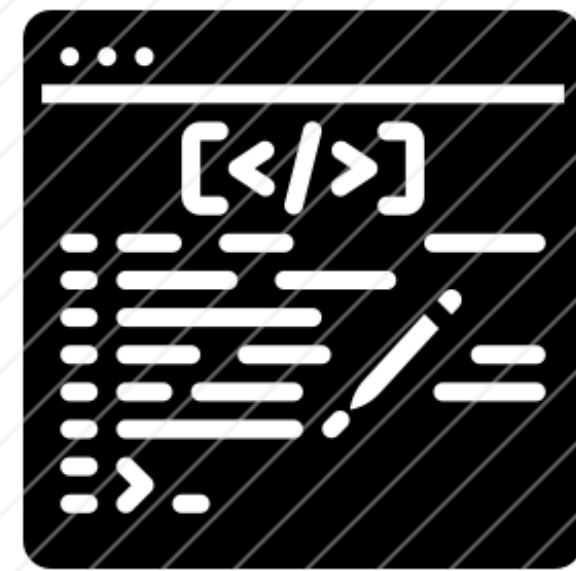
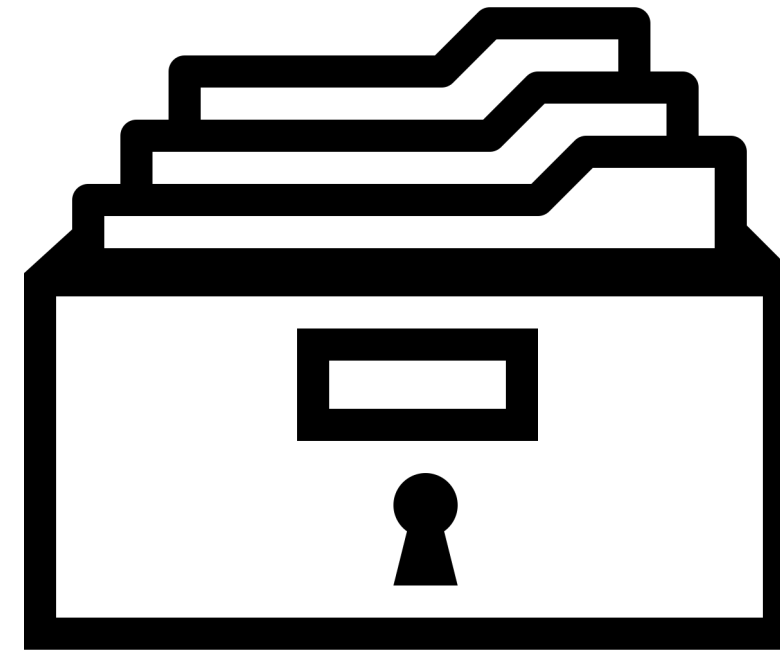
Compilation/Installation changes



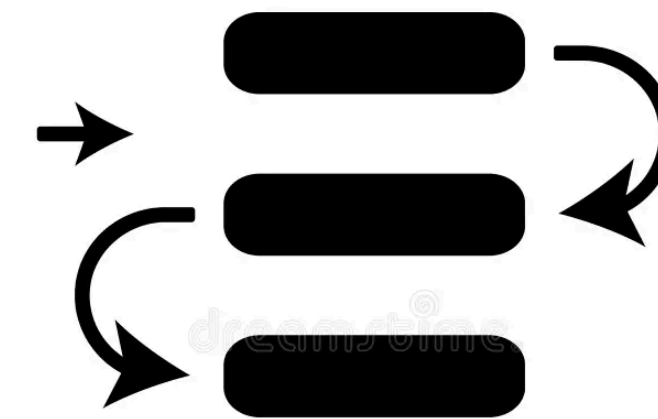
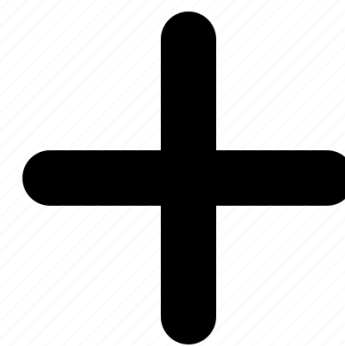
# Software Preservation : Why ?

---

## Archiving



CODE

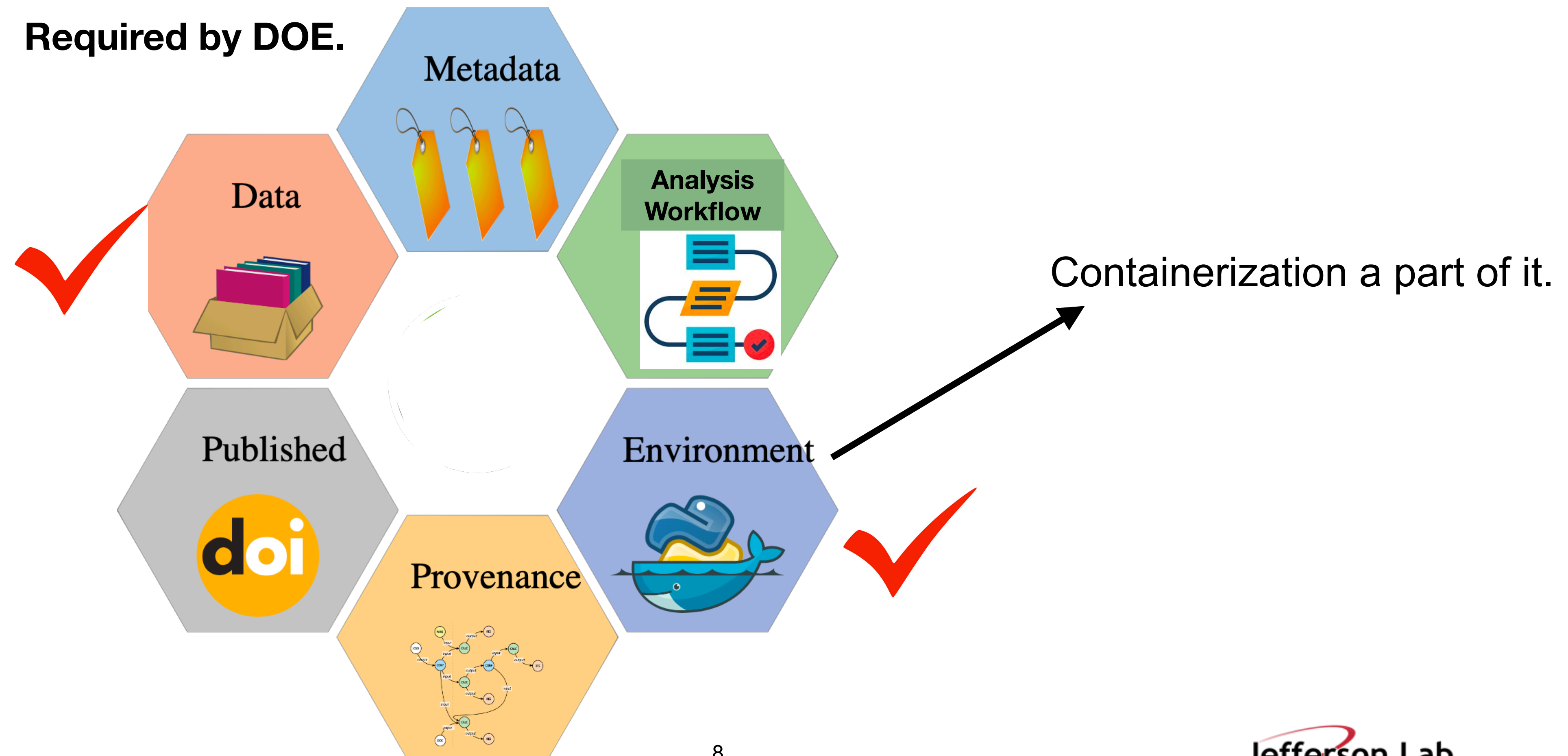


DEPENDENCIES

# Software Preservation : Why ?

## Analysis Preservation

- You can check the mark for software preservation.
- One more step towards Data and Analysis Preservation (DAP).
- Required by DOE.





# Software Preservation : Why ?

---

## Portability

- No problem even if the base OS changes.
- Implies can be used in:
  - Farm at experiment hall.
  - JLab and other clusters/Grid.
  - On your personal computers.



# Software Preservation : How ?

---

- Archiving the code.
  - Releases/Tags tarball in GitHub.
- Containers.
  - Providing the OS, Dependency and compiled software.
- Container registry.
  - Central place to access the containers.
- Automatic containers creation and deployment.
- Tutorial/Instruction Documentations.

# Containers

- Standardized packaging for software:
  - Code
  - Dependencies
  - Host OS
- Encapsulates the entire software ecosystem
- Provides Ready to use software
- Some popular containers are:



# Containers Platform at Jlab

---

- JLab/SciComp is actively developing some 'containerization' best-practices templates.
- Infrastructure to directly support Containerized workflows.
- Works are in progress, will have a announcement soon.



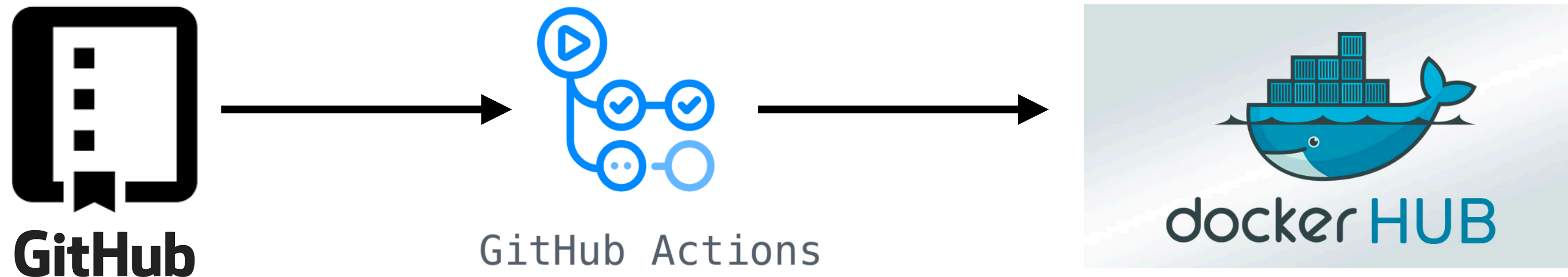
podman



# Automated Container creation

---

- Automatic Build of docker Image using GitHub Action.
- No Human intervention needs.
- Reduced operational/maintenance cost.
- Triggers on New release.
  - **Release version -> docker tag**

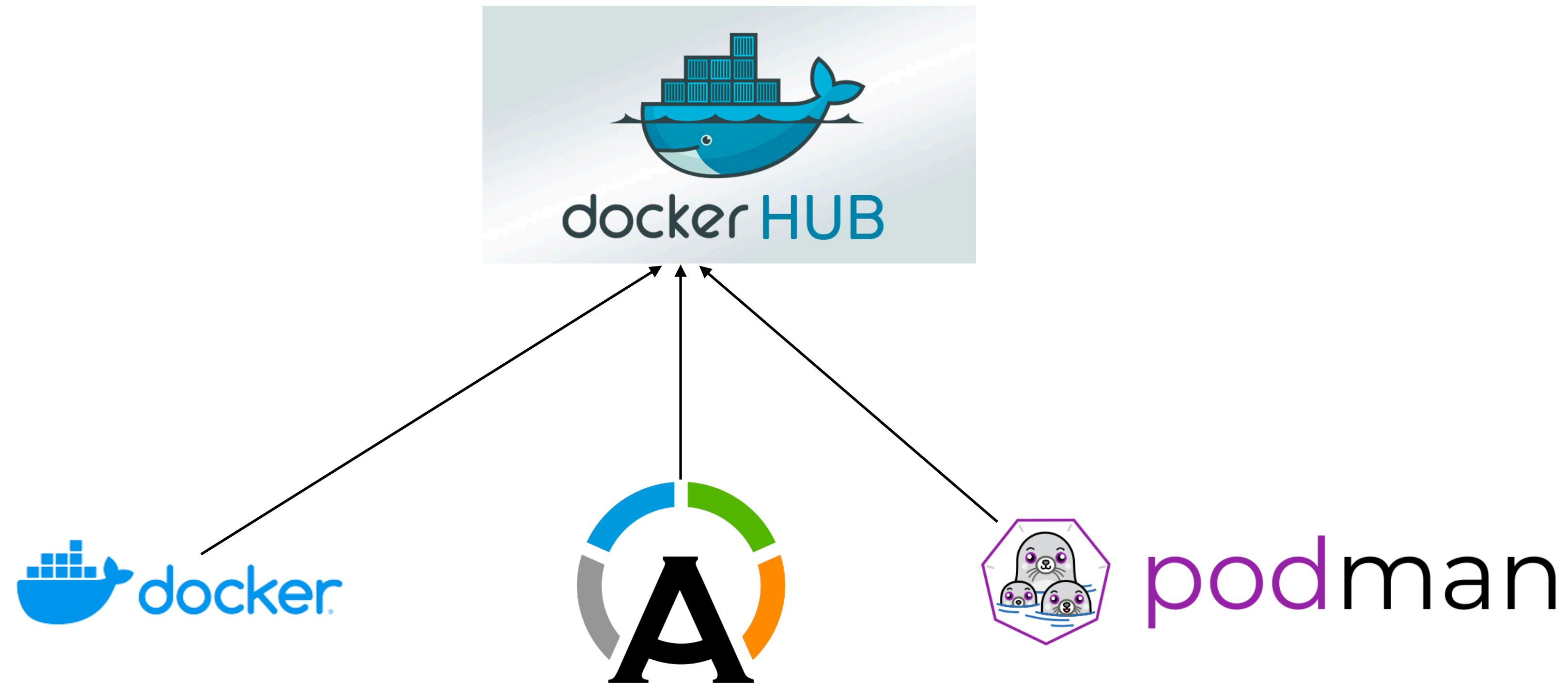


Note:  
dockerHUB != docker  
dockerHUB is a container image registry

# Multi platform support

---

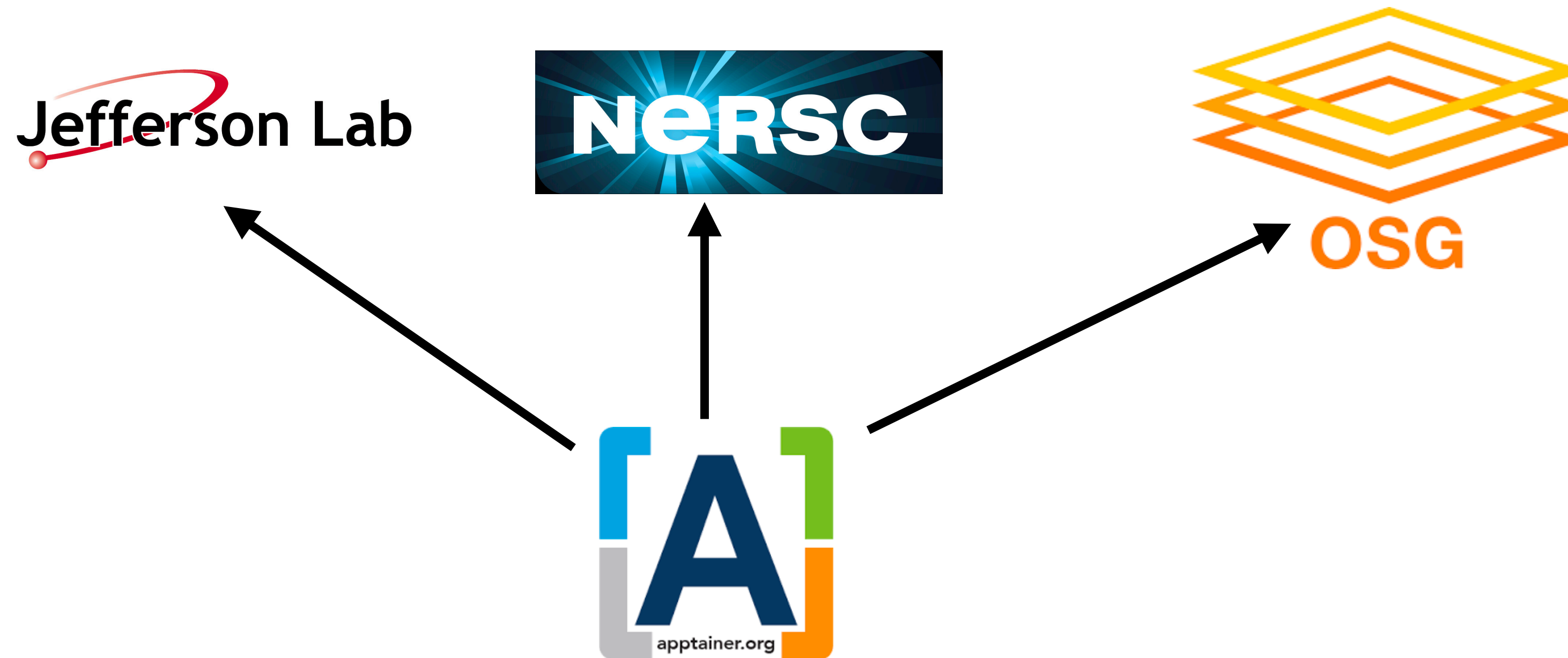
- Same image useable in all three platform.
- No need of extra support.



# External resources

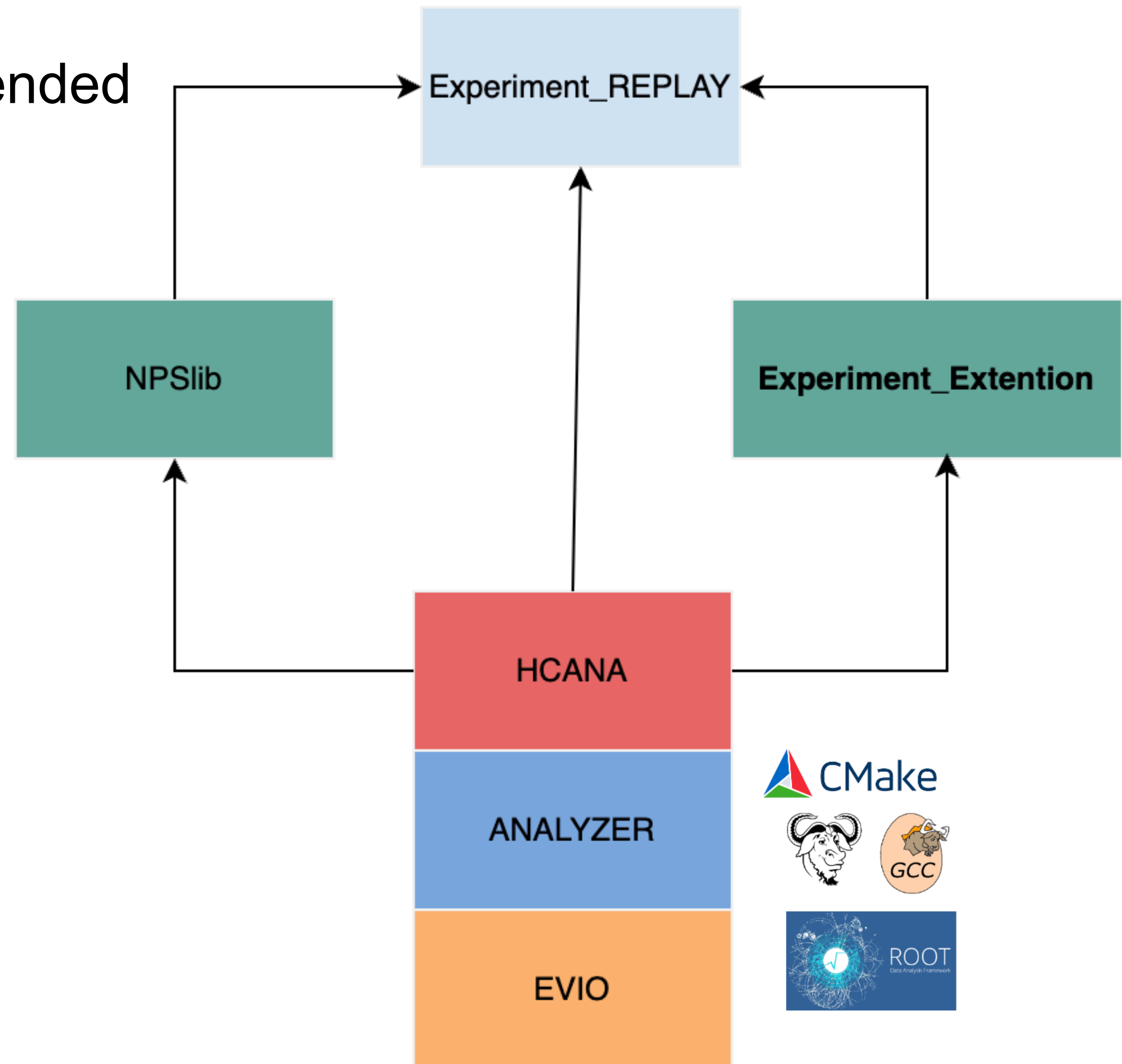
---

- Will be easy too send to External compute center.
- Can utilize the CPU time beyond JLab



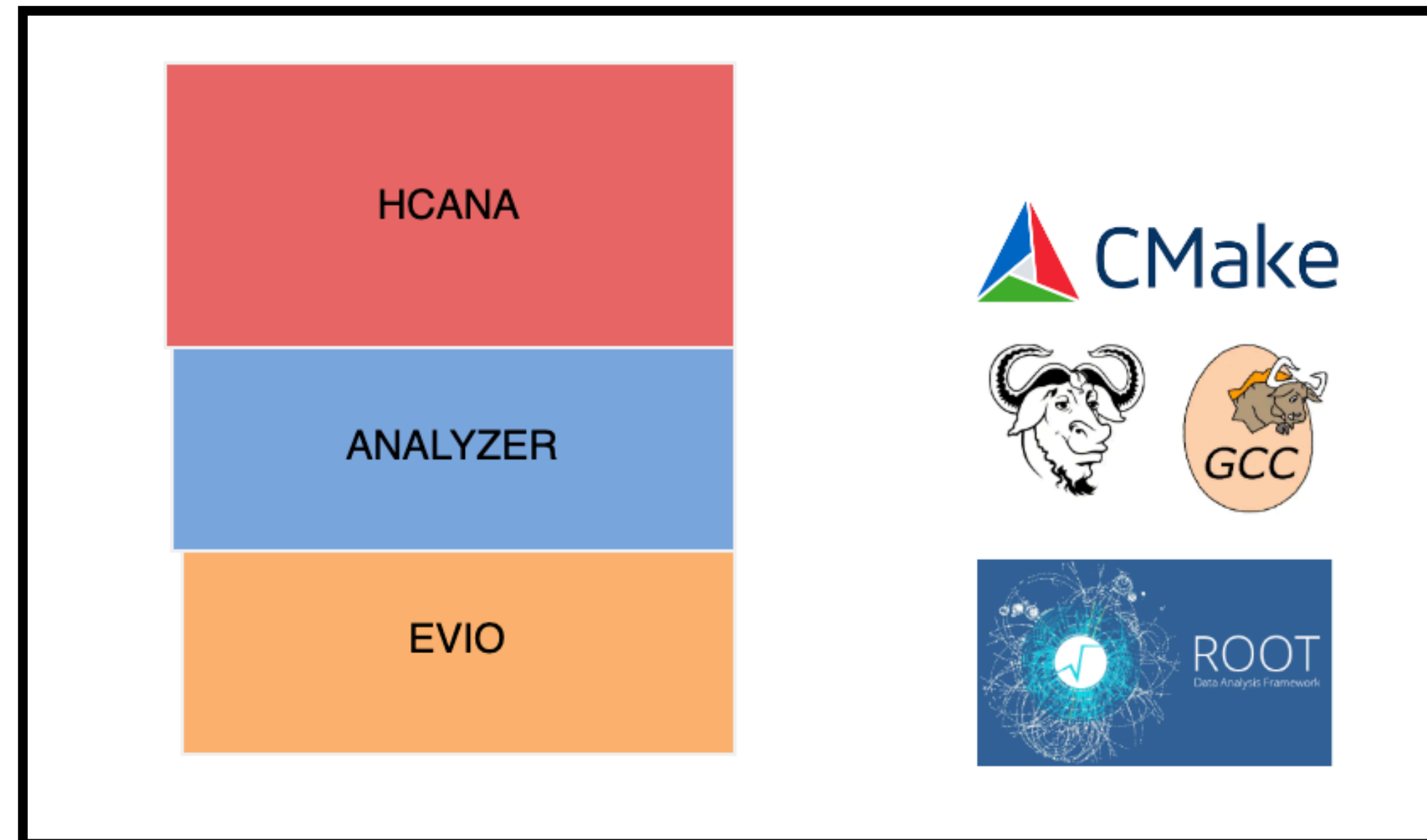
# Hall C Analysis Software stack and dependencies

- HCANA is a core software.
- Some experiments have extended libraries.
  - In different repo.
- Dependencies include:
  - HallA Analyzer
  - EVIO
  - ROOT
- Compilation needs:
  - cMake
  - gcc





# HCANA container:



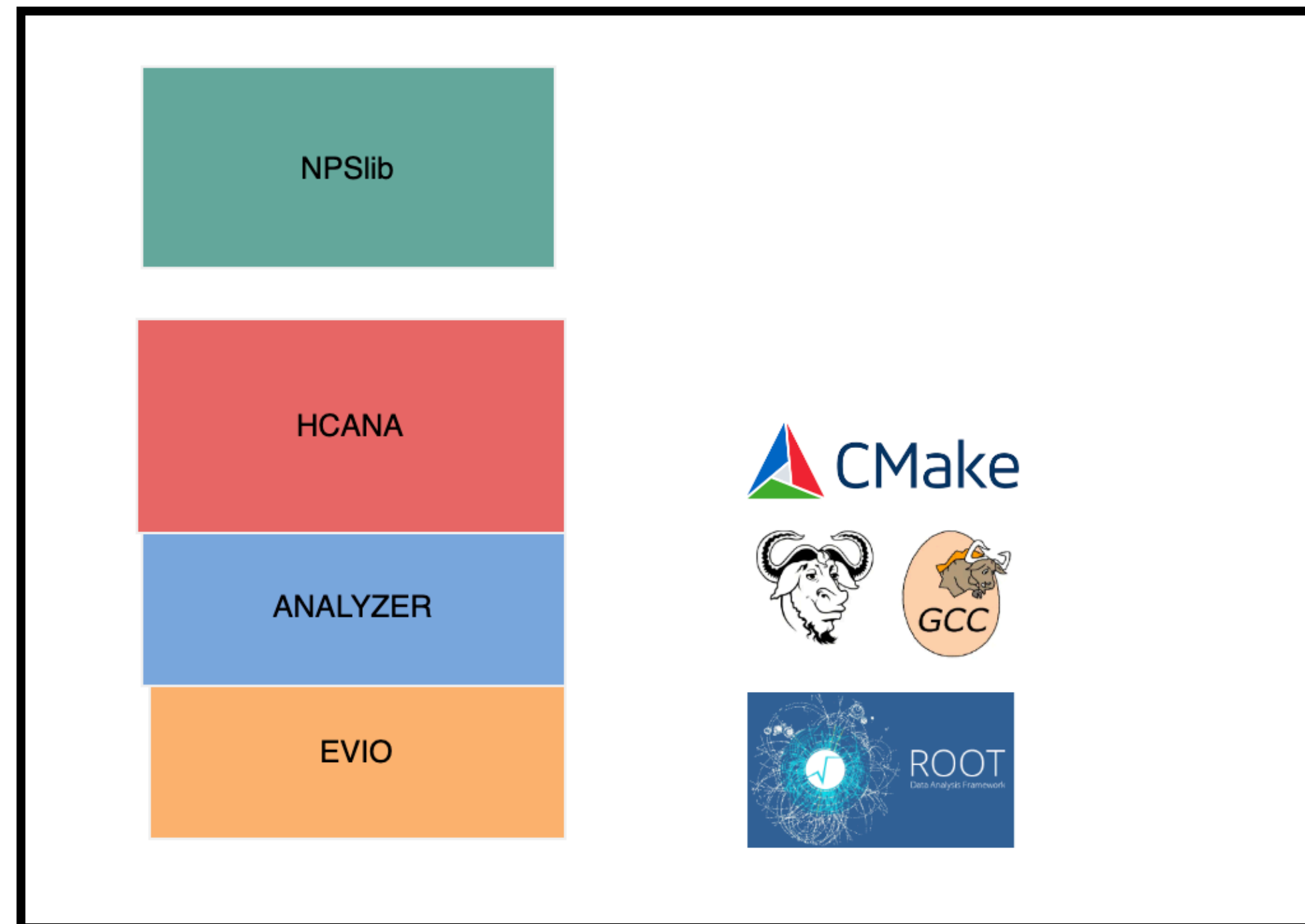
The screenshot shows the Docker Hub repository page for 'jeffersonlab / hallc-hcana'. The repository is described as 'Hall C Analyzer' and was last pushed 2 months ago. The page includes a 'Tags' section with a table of tags and a 'See all' link.

Tag	OS	Type	Pulled	Pushed
hcana-0.98	linux/amd64	Image	12 days ago	3 months ago

<https://hub.docker.com/repository/docker/jeffersonlab/hallc-hcana/general>

- All dependencies and environment variables set.
- Start on analysis.

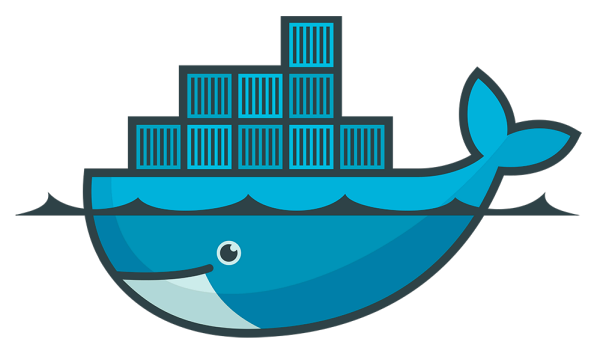
# HCANA with Experiment Extension library: NPSlib



jeffersonlab / hallc-npslib

Description  
NPSLib of hcana

Last pushed: 2 months ago



Tags

IMAGE SECURITY INSIGHTS INACTIVE [Activate](#)

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
v0.1.0		Image	2 days ago	2 months ago

[See all](#)

<https://hub.docker.com/repository/docker/jeffersonlab/hallc-npslib/general>

- All dependencies and environment variables set.
- Start on analysis.

# Getting started: Doc and Tutorial

- Doc for Hall A/C containers made:
  - [https://panta-123.github.io/hcana\\_container\\_doc/html/index.html](https://panta-123.github.io/hcana_container_doc/html/index.html)
- Gives:
  - Introduction, Using via docker/apptainer, Example replay (farm running from terminal and using swif with apptainer image,

The image shows two side-by-side screenshots of a web page for 'Hall A/C container' documentation. The left screenshot is a dark-themed sidebar with a blue header containing a home icon and the text 'Hall A/C container'. Below the header is the Jefferson Lab logo with the tagline 'Exploring the Nature of Matter'. A search bar labeled 'Search docs' is present. Under the heading 'CONTENTS:', a list of topics is shown: 'What is Container?', 'Hall C Analyzer (hcana) Container', 'Docker/Podman Setup', 'Apptainer/Singularity Setup', 'Example', 'NPS REPLAY with apptainer', 'SBS offline and related replay', and 'g4sbs and container'. The right screenshot is a light-themed main content area. It has a breadcrumb trail: a home icon followed by '/ Welcome to Hall A/C container's documentation!'. Below this is a large heading 'Welcome to Hall A/C container's' and a sub-heading 'Contents:'. A list of items follows: 'What is Container?' (with sub-items: 'Key Characteristics of Containers', 'Challenges of Software Package Management', 'Containerization for Software Packages'), and 'Hall C Analyzer (hcana) Container' (with sub-items: 'Introduction', 'Docker Hub Repository', 'Contents', 'Ways to Use the hcana Container', 'Conclusion').

