Joint Institute for Nuclear Research

JINR Container Distribution Service

Nikita Balashov

The 9th International Conference "Distributed Computing and Grid Technologies in Science and Education" (GRID'2023)

7 July 2023

Containers Usage in HEP

- Major motives
  - Reproducibility
  - Portability
  - Support efforts
- Uses-cases
  - Batch job processing on local clusters, Grid and supercomputers
  - CI/CD pipelines
  - Interactive computing systems and personal machines

Base images

Size: Megabytes

Bare OS, may include lightweight infrastructure tools

Experiment images

Size: Gigabytes

Includes all the software stack

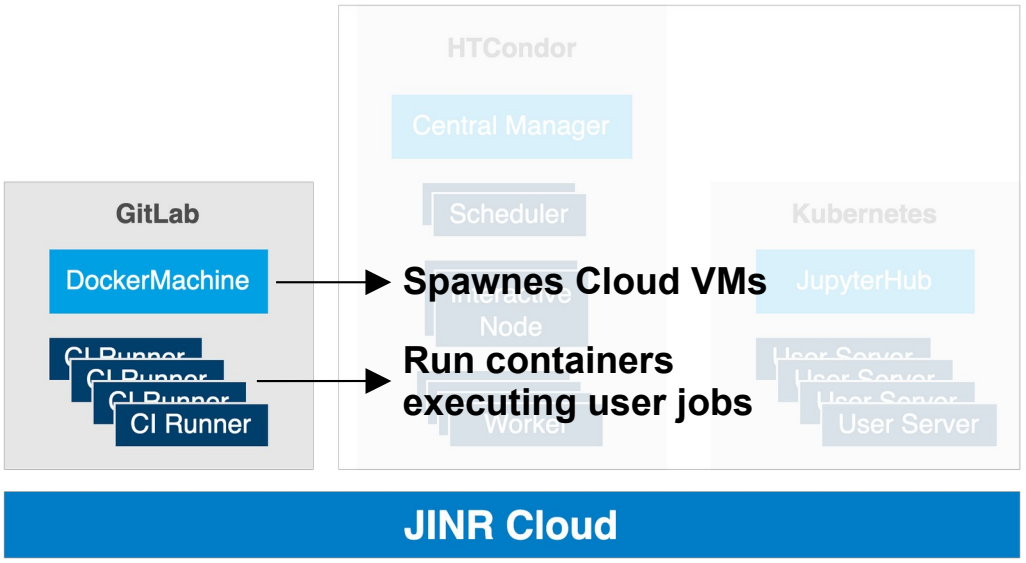
User images

Size: Gigabytes

Usually, based on experiments' images including small user additions

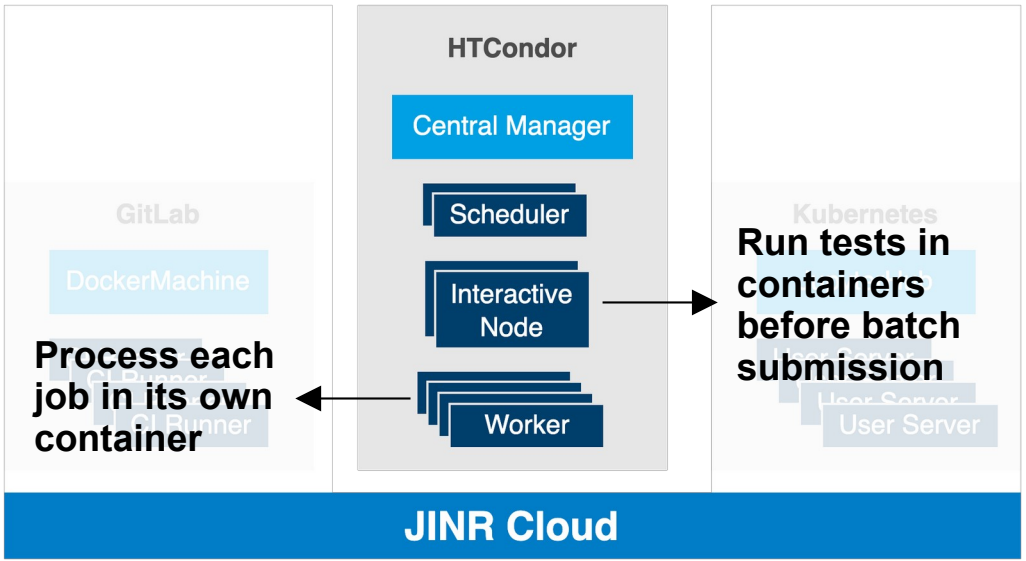
Containers Usage at JINR

- **CI/CD jobs in GitLab:**
  - Docker/DockerMachine
  - Runners are VMs dynamically spawned in JINR Cloud
- HTCondor batch cluster:
  - Apptainer/Singularity (Docker supported, but not widely used)
  - User job are executed in containers
- JupyterHub:
  - Kubernetes/Docker
  - Modified Datascience-notebook image from Jupyter Notebook Data Science Stack for running user servers
- Interactive usage of Docker and Apptainer containers on personal devices and cloud VMs



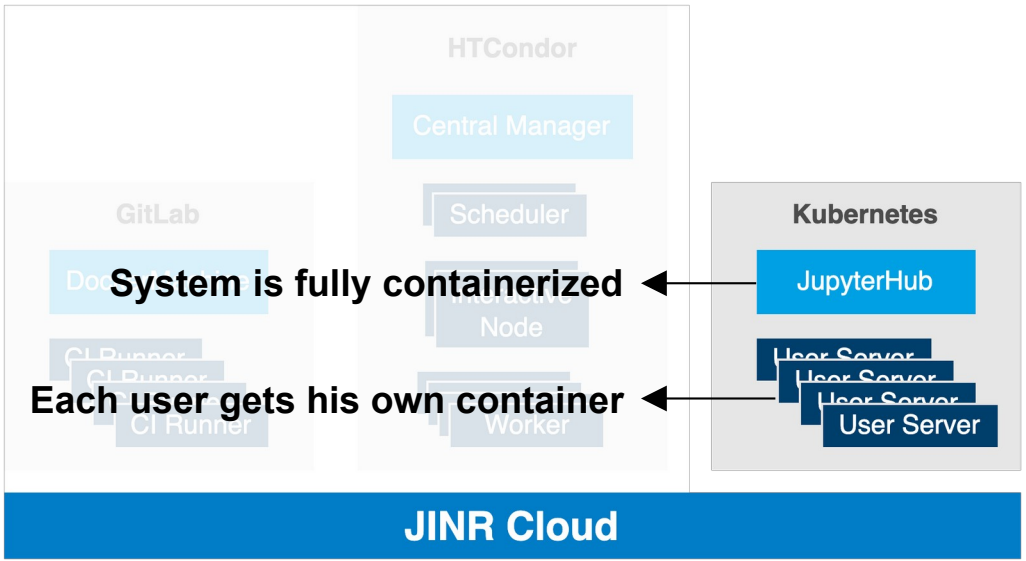
Containers Usage at JINR

- CI/CD jobs in GitLab:
  - Docker/DockerMachine
  - Runners are VMs dynamically spawned in JINR Cloud
- **HTCondor batch cluster:**
  - Apptainer/Singularity (Docker supported, but not widely used)
  - User job are executed in containers
- JupyterHub:
  - Kubernetes/Docker
  - Modified Datascience-notebook image from Jupyter Notebook Data Science Stack for running user servers
- Interactive usage of Docker and Apptainer containers on personal devices and cloud VMs



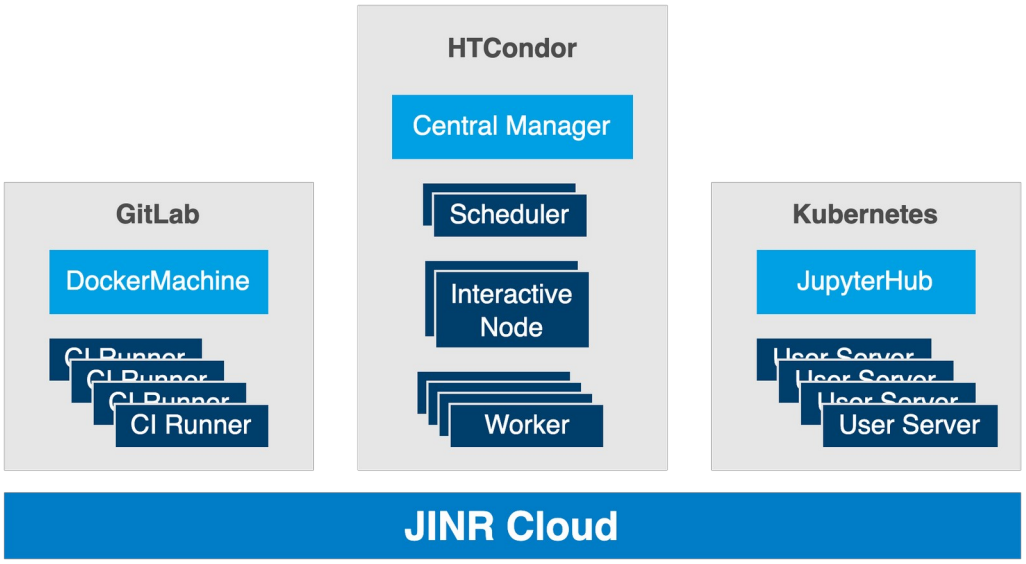
Containers Usage at JINR

- CI/CD jobs in GitLab:
  - Docker/DockerMachine
  - Runners are VMs dynamically spawned in JINR Cloud
- HTCondor batch cluster:
  - Apptainer/Singularity (Docker supported, but not widely used)
  - User job are executed in containers
- **JupyterHub:**
  - Kubernetes/Docker
  - Modified Datascience-notebook image from Jupyter Notebook Data Science Stack for running user servers
- Interactive usage of Docker and Apptainer containers on personal devices and cloud VMs



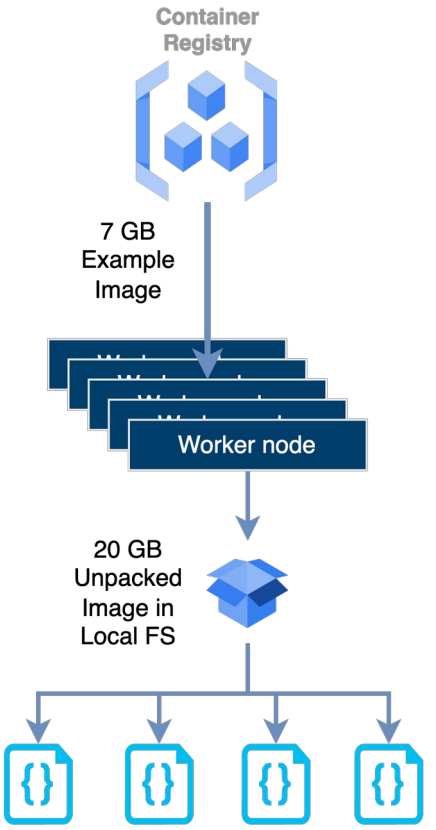
Containers Usage at JINR

- CI/CD jobs in GitLab:
  - Docker/DockerMachine
  - Runners are VMs dynamically spawned in JINR Cloud
- HTCondor batch cluster:
  - Apptainer/Singularity (Docker supported, but not widely used)
  - User job are executed in containers
- JupyterHub:
  - Kubernetes/Docker
  - Modified Datascience-notebook image from Jupyter Notebook Data Science Stack for running user servers
- **Interactive usage of Docker and Apptainer containers on personal devices and cloud VMs**



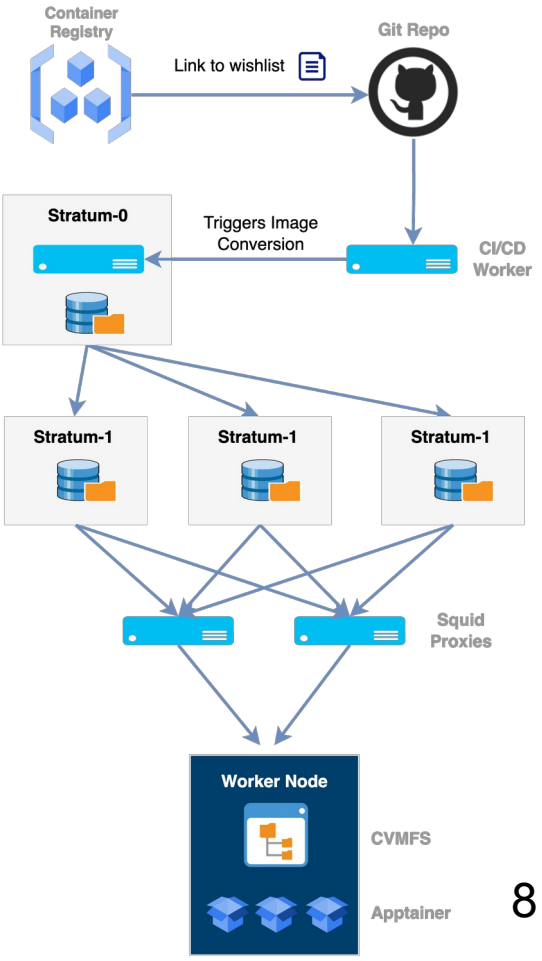
Container Distribution

- Container images can be large and versioned, e.g. borexino-ui:
  - In registry: ~7 GB
  - Unpacked on the worker: ~20 GB
- Images are traditionally stored in so called registries
  - At JINR we use GitLab Registry (S3 backed with Ceph)
- Main challenges:
  - Images need to be synced to hundreds of nodes posing huge load on image storage and network
  - Client storage is usually limited and needs to be reclaimed when an image is not used



HEP Approach Based on CVMFS

- Traditional registries are used to store docker images
- Images are listed in a GitHub/GitLab project
- CI jobs unpack images and store them in CVMFS repository
- Clients run containers from CVMFS “folders”
  - Lazy loading with per file level of caching granularity
  - Automatically reclaims storage space
- Apptainer is natively supported
- Some experimental support of Docker, containerd and Podman



Available Solutions

- Solution from CERN
  - Public image list on GitHub: https://github.com/cvmfs/images-unpacked.cern.ch
  - Private image list on CERN GitLab
  - CVMFS DUCC used for image conversion and publishing
  - CVMFS repository: unpacked.cern.ch
- Solution from Open Science Grid (OSG)
  - Public image list on GitHub: https://github.com/opensciencegrid/cvmfs-singularity-sync
  - Custom set of python scripts to convert and publish images
  - CVMFS repository: singularity.opensciencegrid.org
- Restrictions of self-hosted installations
  - CVMFS server needs docker installed and running
  - Escalated privileges on the CVMFS server for publisher
  - Increased load on CVMFS server when no additional publishers available

CERN Wishlist

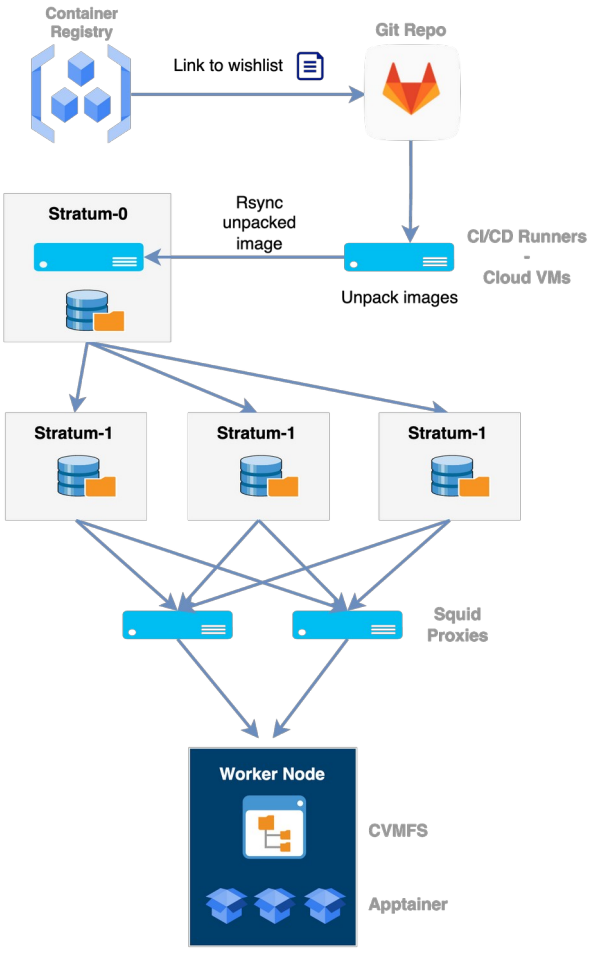
```
version: 1
user: cvmfsunpacker
cvmfs_repo: 'unpacked.cern.ch'
output_format: 'https://gitlab-registry.cern.ch/unpacked/sync/$(image)'
input:
    - 'https://registry.hub.docker.com/library/centos:centos7'
    - 'https://registry.hub.docker.com/engineren/ilcsoft:test'
    - 'https://registry.hub.docker.com/uegede/weather:latest'
```

OSG Wishlist

```

# This file is a list of Docker images to synchronize to singularity.opensciencegrid.org.

# First, some generic CentOS images:
centos:latest
centos:centos6
centos:centos7

# Fairly common Linux distros
debian:latest
debian:stable
debian:testing
debian:unstable
ubuntu:latest
fedora:latest
rockylinux:8
```

JINR Solution

- Based on modified OSG software
- Does not require any additional CVMFS server configuration and privileges
- Conversion can be executed on CI/CD nodes
- Containers are stored in containers.jinr.ru cvmfs repo via https://git.jinr.ru/cvmfs-container-sync/registry
- Solution can be used to setup private conversion service to any private repository



Conclusions

- Public moderated services from OSG and CERN can be used by to distribute containers via CVMFS
- Presented JINR Container distribution service can be used by JINR employees and participating parties
- JINR solution can also be used as an example to organize a personal publishing service
- Implemented solution is only suitable for Apptainer/Singularity, but we plan to experiment with support of other runtimes

Thanks!