

Integration between container and CVMFS

...

The narrative

CVMFS solved the problem of software distribution.

Containers solved the problem of having a reproducible environment.

Serve containers over CVMFS bring the best of the two worlds and it is possible thanks to Nikola.

Provide a complete solution for users, that will survive without me

Highlights

- `unpacked.cern.ch`
- DUCS
- CVMFS Ingest command

Highlights

- `unpacked.cern.ch` <- An home for container images
- DUCC
- CVMFS Ingest command

Highlights

- `unpacked.cern.ch` ← An home for container images
- `DUCC` ← Manages and keeps unpacked clean
- `CVMFS Ingest` command

Highlights

- `unpacked.cern.ch` ← An home for container images
- `DUCC` ← Manages and keeps unpacked clean
- `CVMFS Ingest` command ← Allows `DUCC`

From the bottom down

unpacked.cern.ch

- Users need a repository where to find the images
- /cvmfs/unpacked.cern.ch
 - Store Docker layers
 - Store Singularity rootfs
- Hierarchical structure to manage the complexity and the subcatalogs

DUCC

- **Daemon to Unpack Containers in CVMFS**
 - Thanks to OpenScienceGrid
- **Declarative approach to provide containers images in a CVMFS repository**
- **Support Docker, Singularity & proof of concept with containderd**

CVMFS Ingest command

- Allows to ingest a tarball directly inside CVMFS
- Does not touch the scratch filesystem
- Useful to work with containers
 - Containers are distributed as set of tarballs

Ensure smooth operations

- **unpacked.cern.ch**
 - Use service user
 - Run on Jenkins
- **DUCC**
 - Integrate inside main CVMFS repository
- **Ingest**
 - Integrate inside `cvmfs_server`

Possible future works

- Support for multiple architecture
- Enhance support for containerd

Thanks!