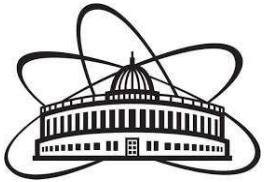
Using containerization for BM@N distributed data processing

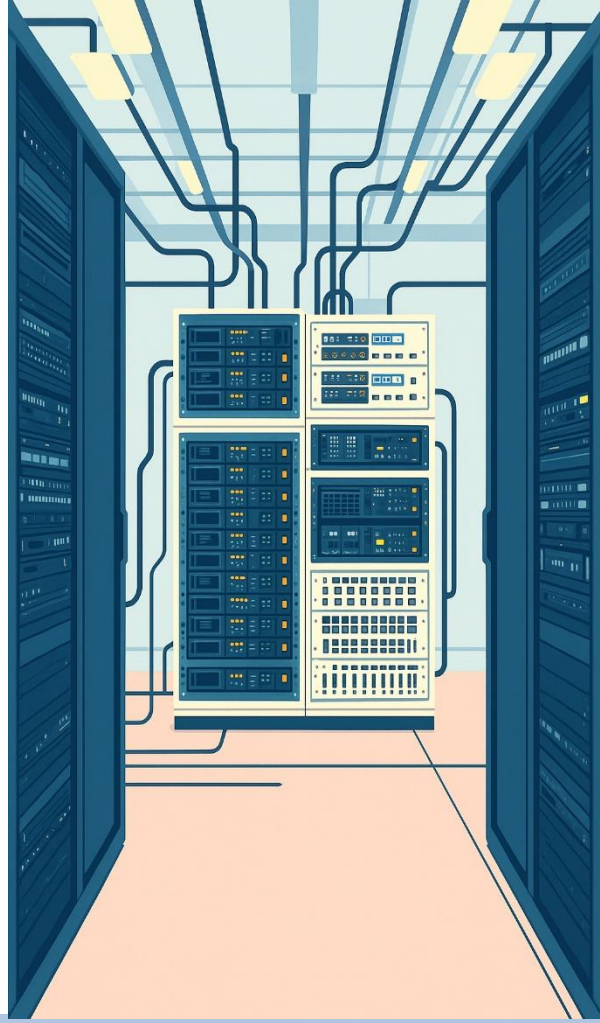
Rinat Nizamov (SPbU)
K.V. Gertsenberger (JINR)
S.A. Nemnyugin (SPbU)



JINR Cluster Infrastructure

- Heterogeneous platform “HybriLIT” (OS: CERN CentOS 7)
- JINR CICC complex (OS: Scientific Linux 7.9)
- NICA, NCX Cluster (OS: CentOS Linux 7)
- BM@N DAQ Computing Center (OS: AlmaLinux 9)
- JINR Cloud Infrastructure

Various Linux distributions and systems with different tool versions are used. A unified solution is needed.



Simplifying Data Processing

CVMFS

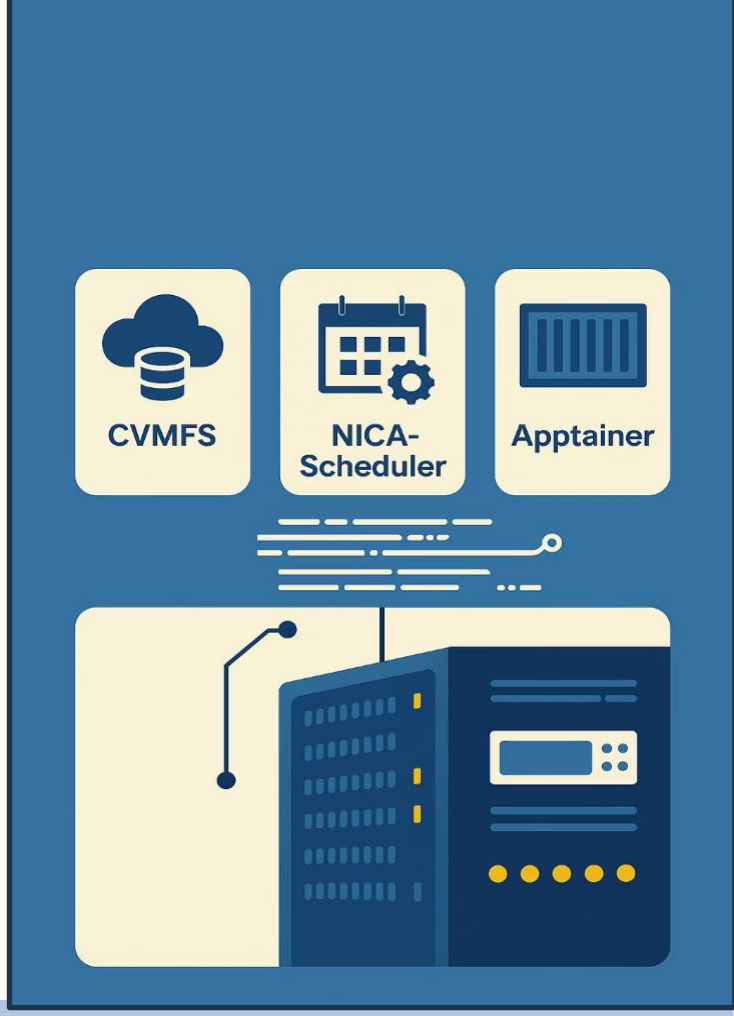
Unified software repository — software is installed centrally and automatically available on all clusters via */cvmfs* directory

NICA-Scheduler

Simplified job launching — one configuration for different clusters and queue systems (SGE, SLURM, etc.)

Apptainer

Environment containerization — ensures compatibility and reproducibility of analysis across any computing nodes



Distributing Software via **CernVM-FS**

CernVM-FS

CernVM-FS

A network file system developed at CERN for software distribution via HTTP with caching

How it works

A network file system developed at CERN for software distribution via HTTP with caching.

Usage at JINR

The */cvmfs/bmn.jinr.ru* repository provides unified software (FairSoft, FairRoot) to all nodes of NICA, HybriLIT, etc

Containerization with **Apptainer**

Apptainer

Solves OS and environment dependency issues, provides portability and task isolation **without root access**.

Working Principle

SIF format containers run as regular processes, using cluster resources directly, **without performance loss**

Guarantees

consistent execution across all clusters (CentOS, Scientific Linux, etc.); simplifies distributed work and makes it predictable

NICA-Scheduler: for executing tasks on multi-core machines and clusters in parallel

NICA-Scheduler

A module developed for the BmnRoot (and MpdRoot) framework. Automates job execution for experimental data processing on clusters. Compatible with SLURM, SGE, Torque

How it works

The user creates an XML file describing what, where, and how to process → NICA-Scheduler splits it and sends parts to the queueing system

Convenient Web Interface:

Allows graphical configuration of jobs in a browser (add macros, files, parameters), further lowering the entry barrier

Architecture

Acts as a middleware layer between users and job schedulers, embedded in the experiment's framework

New “**container**” execution mode

New “container” execution mode in NICA-Scheduler

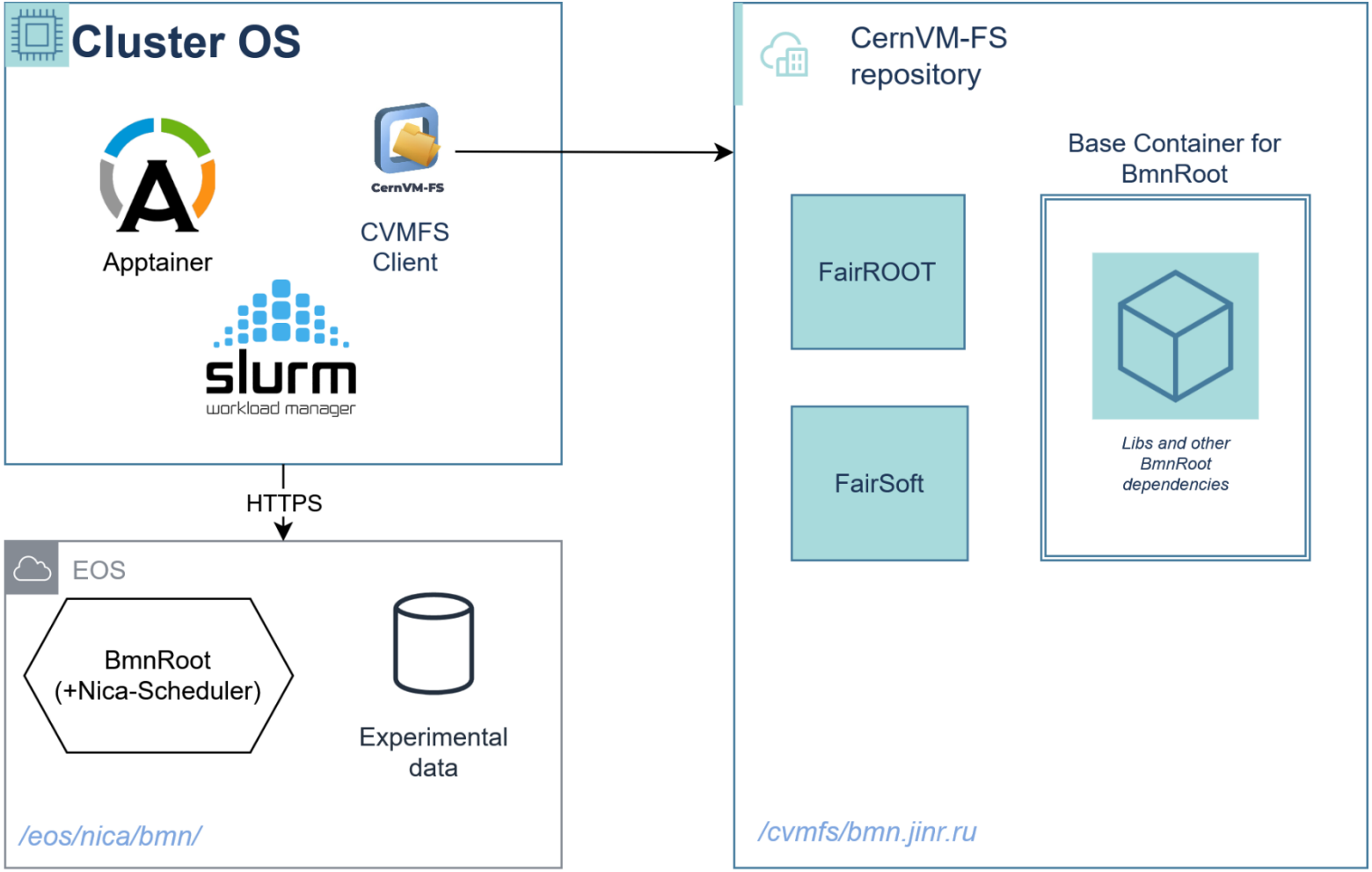
Allows launching jobs inside an Apptainer container with a ready-made environment from the NICA-Scheduler module (e.g., for BmnRoot)

How it works

Instead of a regular launch — uses a container image (.sif) from CVMFS. Everything is controlled via the XML file with the “container” mode

Unified Environment in Container

In this mode, the container environment is preconfigured and remains unchanged regardless of the cluster system, as it is taken directly from the official CVMFS repository


Cluster OS
Apptainer
CernVM-FS
CVMFS Client
slurm
workload manager
HTTPS
EOS
BmnRoot (+Nica-Scheduler)
Experimental data
/eos/nica/bmn/
CernVM-FS repository
Base Container for BmnRoot
FairROOT
FairSoft
Libs and other BmnRoot dependencies
/cvmfs/bmn.jinr.ru

Launch example

bash$: nica-scheduler –d bmn_sim_container_lxui.xml

1. nica-scheduler generates two .sh scripts: **job_{id}.sh** and **apptainer_run.sh** in the root directory (specified via the work_dir tag)
2. job_{id}.sh parses the input files and launches each through the intermediary script apptainer_run.sh in the container environment
3. apptainer_run.sh executes the command: **“root -q –b …”** with passed arguments
4. As a result, each file is processed on the same cluster hardware, but within a different selected container system environment (e.g., AlmaLinux 9 in this case)

Overhead of container launch is quite low (a few seconds), and processing time remains nearly unchanged

```
lxui01:/eos/nica/bmn/users/itsrin8/bmnroot/services/nica_scheduler/examples > nica-scheduler -d bmn_sim_container_lxui.xml
Warning in <UnknownClass::SetDisplay>: DISPLAY not set, setting it to softco-v32.cust.smartspb.net:0.0
DEBUG nica-scheduler$ File count in the regular expression: 99
DEBUG nica-scheduler$ Batch command (length 194): sbatch -J bmn_sim_container  -a 1-99%30 -D /afs/jinr.ru/user/i/itsrin8/scheduler_cont /eos/nica/bmn/users/its
rin8/bmnroot/build/bin/job_3a2d40c4245169e1.sh
DEBUG nica-scheduler$ Batch command output = Submitted batch job 11943230

nica-scheduler$ The job 'bmn_sim_container' has been submitted with ID: 11943230. Enter 'squeue' command to check status.
lxui01:/eos/nica/bmn/users/itsrin8/bmnroot/services/nica_scheduler/examples > squeue -u itsrin8
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
11943230_[31-99%30      cicc bmn_sim_  itsrin8 PD       0:00      1 (JobArrayTaskLimit)
```

Conclusion

Achievements:

- Unified containerized environment for BM@N data processing with Apptainer.
- Automated job execution using NICA-Scheduler with container support

Benefits :

- Consistent environments across all clusters.
- Reproducible and predictable data processing.
- Minimal performance overhead

Thank you for your attention!