

# Containerization in the ATLAS Offline Code Management System

---

Chris Lee, Asoka De Silva, Alexander Undrus, Shuwei Ye



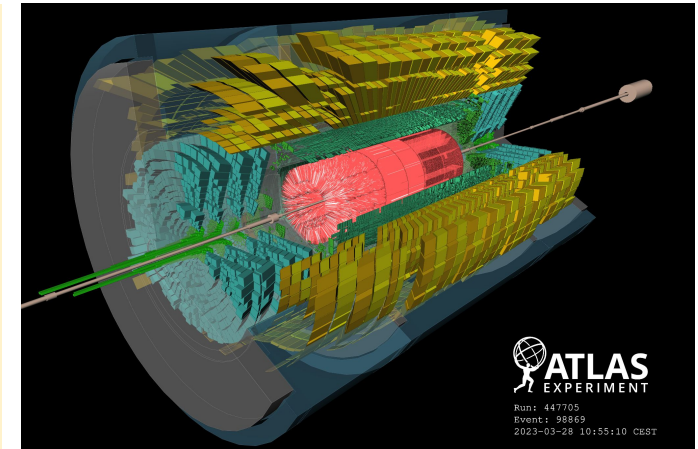
CHEP 2024 Conference  
October 19 -25, 2024



The ATLAS detector is the largest volume detector at the Large Hadron Collider (LHC) at CERN (the European Organization for Nuclear Research) in Switzerland.



The ATLAS Collaboration consists of over 6000 members from more than 180 institutions in 42 countries. ATLAS takes experimental physics into unexplored territories – searching for new processes and particles that could change our understanding of energy and matter.



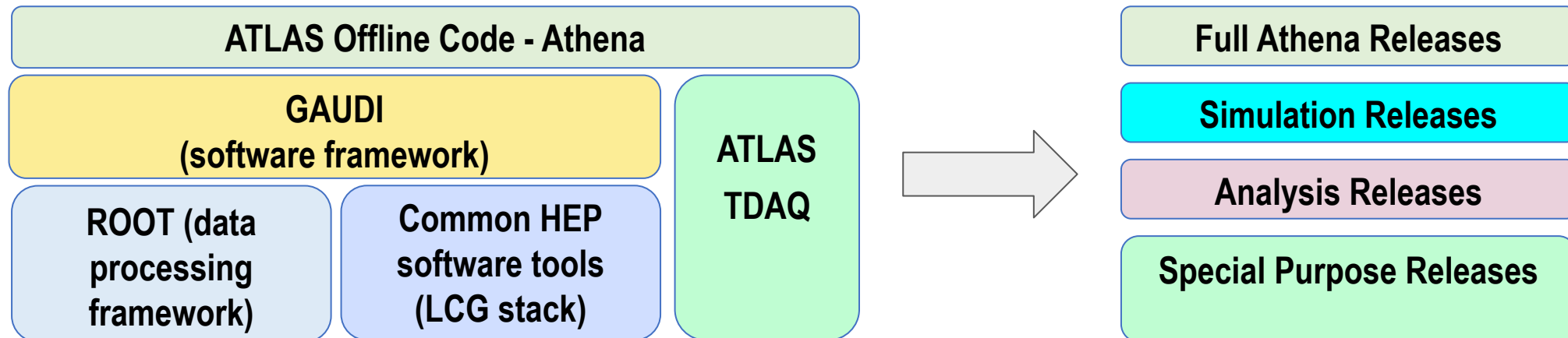
ATLAS currently collects about 10 PB of new raw data per month, and uses over 130 computing centers worldwide for processing and analysis. The ATLAS custom software totals more than 5 million lines of code. It converts detector signals into information that physicists can study.

# This Talk

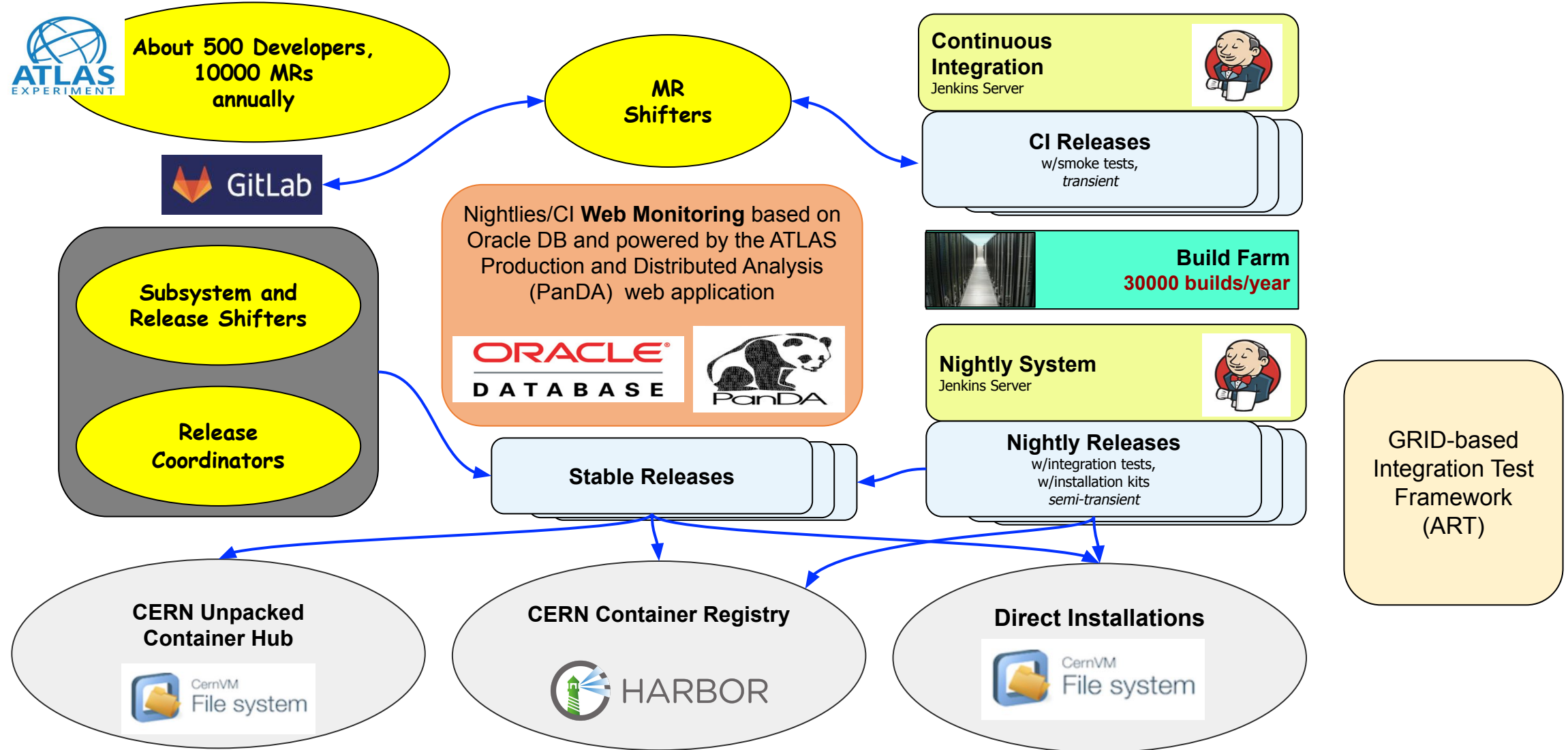
- **Outlines the role of containerization technology in the ATLAS offline software development infrastructure**
- **Describes the utilization of container technology in the Nightly Build System**
- **Details storage solutions for software release images**
- **Explains methods for setting up runtime environments**

# ATLAS Offline Code Base: Big Scale and Complexity

- Over 5 million lines of C++ and Python code
- Public under an APACHE 2.0 license
- Event generators, Simulation, Reconstruction, Calibration, Analysis software
- Under constant development to get the latest technologies, analysis techniques
- Require over 200 external packages (mostly supplied by CERN SFT team, ATLAS TDAQ releases, GAUDI architecture framework, generators)
- Full and project-specific (e.g. Analysis) releases are available



# Offline Software Development Workflow at a Glance



# Containers in ATLAS Software: History

- **ATLAS began adopting container technologies soon after Docker's release in 2013**
- **2017: R&D initiative centered on container technologies for software development**
- **2020-2021: Activity of the Containerization Task Force**
  - Developed a pipeline for building software release containers
  - Enabled efficient utilization of HPCs in ATLAS production campaigns
  - Defined containers as a crucial component for running ATLAS workflows at opportunistic sites
- **2022-2023: Expansion of container utilization, support, services**
  - Containers had been fully integrated into analysis workflows
  - Custom container images were created for Machine Learning workflows and integrated into the Analysis Infrastructure across ATLAS Tier-2 and Tier-3 centers
  - User-friendly Harbor Container repository was launched at CERN
  - CVMFS-based Hub of unpacked images for efficient distribution to remote sites was established
- **2024: Integration into the Nightly System**
  - Introduced builds within historical OS containers to support legacy workflows
  - Image builds have been added in selected nightly jobs

# Roles of Container Technology in the Nightly System

## Builds within containers

- Support 12 legacy nightly branches built within CentOS7 containers running on EL9 machines
  - Important for preservation of legacy workflows in view of CentOS7 phase-out in 2024
- Special mainstream nightly jobs within EL9 containers, with the primary goal of validating the default EL9 container and associated functionalities
- All releases undergo extensive testing, both locally and across the Grid
- All builds are installed directly (non-containerized) into the CVMFS file system for wide accessibility

## Image builds

- Developers can trigger jobs within the nightly systems to build images for nightly and stable releases
  - A selective option facilitates efficient resource use by building only images that users require
- Images are built on top of a standard OS base image and undergo thorough sanity tests
- Images include tools and libraries from CERN's common software stack (LCG)
- Developers can choose between the full software release image (16GB compressed) or smaller containers for analysis or simulation tasks (2 - 8 GB compressed)

# Storage Solutions



- **CERN's Harbor container registry**

- Powerful open-source container image registry solution
- Advanced capabilities for security, management, and replication
- Sufficient capacity to store nightly releases with a 7-day retention policy and a robust collection of stable releases for longer-term storage

- **CERN's "unpacked" CVMFS storage hub**

- Delivers rapid distribution of unpacked Docker images across the Grid, optimized for frequent pulls by numerous worker nodes
- Container layers are extracted into directory tree (flat root filesystem), leveraging CVMFS advanced file-based deduplication
- CVMFS ensures optimal caching at the client side, fetching only the necessary files rather than the entire image, resulting in significant performance gains
- Primarily used for stable release containers used in production Grid jobs



# Image Naming Conventions

- **Image names** include identifiers ensuring clarity about which environment a particular image is for, helping with version management and accessibility via the ATLAS environment setup tools
  - semantic version (in major.minor.patch format)
  - timestamp
  - platform identifier
- **Registry organization**
  - images are organized in folders with names indicating their purpose (e.g., analysis, simulation)

Image name example:

```
25.0.18_2024-09-19T0401_x86_64-e19-gcc13-opt
```

# Containers Startup and Setup

- **Streamlined environment setup** is ensured by a dedicated Linux shell script embedded within the containerized release during the final stage of image building
- Container startup is fully supported and seamlessly integrated with ATLAS environment setup tools

**ATLAS Local Root Base (ALRB)** - centralized tool for setting up the software environment for accessing and using ATLAS releases, ROOT framework and other HEP tools and libraries

- Launches ATLAS release containers with a single command

```
setupATLAS -c  
docker://registry.cern.ch/atlas-nightlies-<project>/<folder>:<image name>
```

**AtlasSetup environment management framework** works alongside ALRB to configure for the ATLAS software suite. It sets paths, environment variables, and libraries needed by ATLAS applications

- Provides lists of available images on CVMFS in friendly and compact format
- Automatically selects and launches the OS container compatible with the requested ATLAS release

# Conclusion

## In the ATLAS Nightly System:

- **Container usage has become an essential component of various build jobs**
- **Container technology helps to preserve legacy workflows by providing a secure and reliable environment for building and testing on historical operating systems**
- **Users can request the creation of images for specific nightly and stable releases**
  - Images are automatically installed into CERN's Harbor Registry
- **Image naming conventions and setup procedures are standardized to align with ATLAS environment management systems**
- **This standardization allows for:**
  - Seamless downloads
  - Streamlined startup with a single command line across multiple architectures and operating systems