Containerization in ATLAS Software Development and Data Production

Nurcan Ozturk, Alessandra Forti, Alex Undrus, Marcelo Vogel

25th International Conference on Computing in High-Energy and Nuclear Physics
May 17-21, 2021

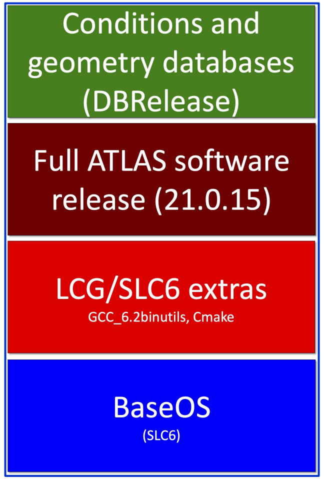
Outline

- Motivation
- Strategy for Container Building and Registration
- Container Distribution
- Containers in Monte-Carlo Data Production
- Container Builds in ATLAS Nightly System
- Container Testing in ATNC Framework
- Summary and Outlook

Motivation

- Containers have a long history in ATLAS starting from 2017 as a R&D activity.
- Operating System (OS) containers and standalone containers for user analyses advanced first.
- Then standalone containers with a single software release built for use of High Performance Computing (HPC) sites with no CVMFS and network access:
  - More complex than analysis containers: must include about 250 external packages from LCG (LHC Computing Grid) stack and conditions, trigger, geometry data.
  - First prototype presented in CHEP 2019 and proven successful thus opened the way to automate the whole process.
- Goal of this work: Develop a pipeline to build software release containers and integrate them in the production system in a more uniform and automated way for both grid and HPC sites.

Strategy for Container Building and Registration

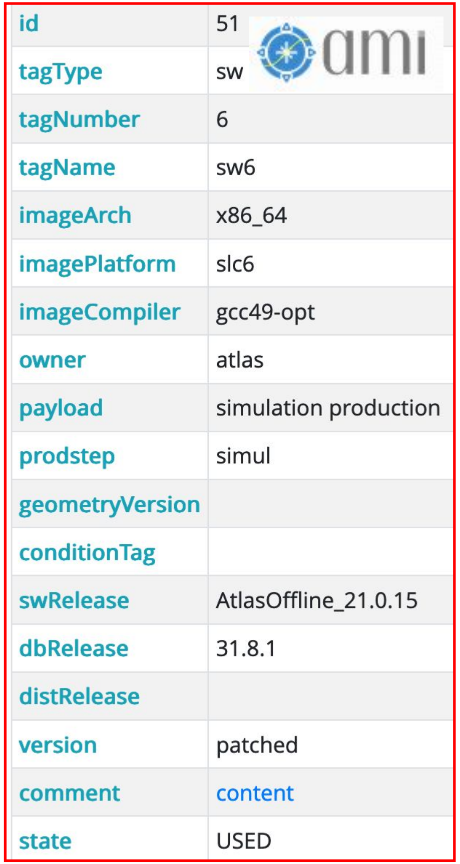

Conditions and geometry databases (DBRelease)
Full ATLAS software release (21.0.15)
LCG/SLC6 extras
GCC_6.2binutils, Cmake
BaseOS
(SLC6)

Container building in layers:

- Two Dockerfiles are used in a continuous integration (CI) pipeline.
- First to install a single software release on top of the matching OS base image and LCG/SLC6 extras.
- Second to install detector conditions, trigger and geometry databases (via a custom-made package called DBReleases). Also generates a python script used to set up the software release and the runtime for payload execution.
- This pipeline creates containers for detector simulation workflow in fully standalone mode, namely no need for CVMFS access or network connection from the host site.

Container registration in AMI (ATLAS Metadata Interface):

- Naming convention following the Open Container Initiative image name specifications: <repository user>/<repository name>:<cacheName>.<AMItag>-<counter>
- Example: atlas/athena:21.0.15.sw6-0
- AMItag: software stack identifier which uniquely identifies how to prepare a software environment for a given ATLAS production workflow. Example: sw6.



id	51
tagType	sw
tagNumber	6
tagName	sw6
imageArch	x86_64
imagePlatform	slc6
imageCompiler	gcc49-opt
owner	atlas
payload	simulation production
prodstep	simul
geometryVersion	
conditionTag	
swRelease	AtlasOffline_21.0.15
dbRelease	31.8.1
distRelease	
version	patched
comment	content
state	USED

Container Distribution

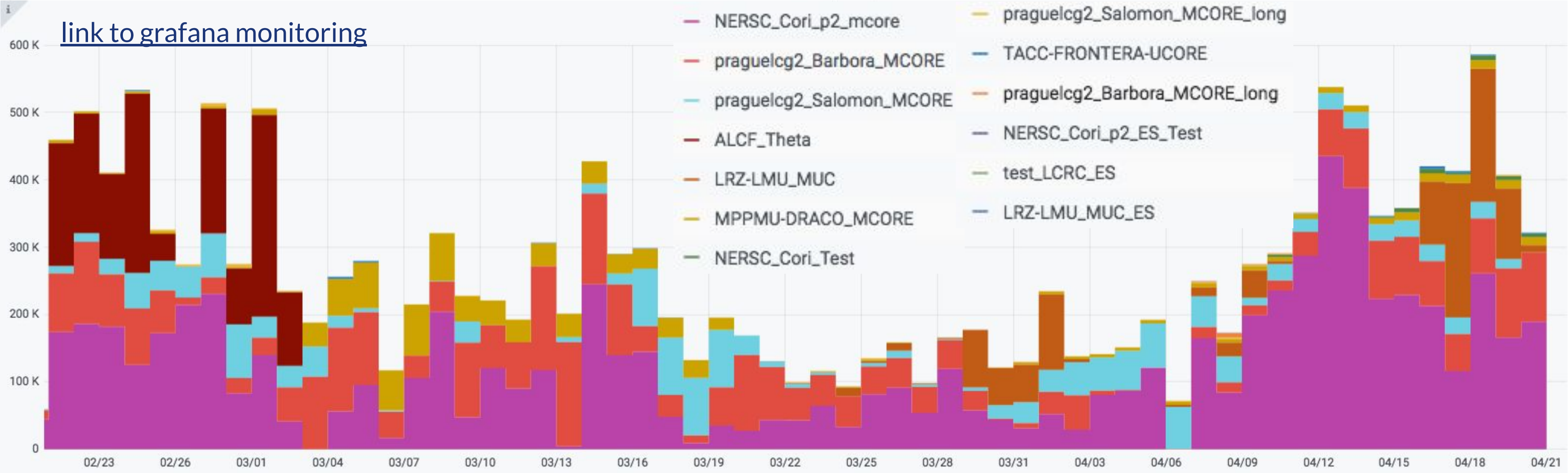
- Containers are uploaded in Docker Hub registry and automatically synchronized to CVMFS under /cvmfs/unpacked.cern.ch/registry.hub.docker.com/atlas.
- In light of recent Docker Hub policy changes ATLAS is moving to GitLab registry.
- CERN-IT is working to integrate Harbor registry with GitLab and CVMFS. ATLAS is contributing to tests.

Containers in Monte-Carlo Data Production

Containers are fully integrated in the ATLAS production system to run in a more uniform way across grid and HPC sites. Standalone release containers are used for Monte-Carlo detector simulation at the following HPC sites:

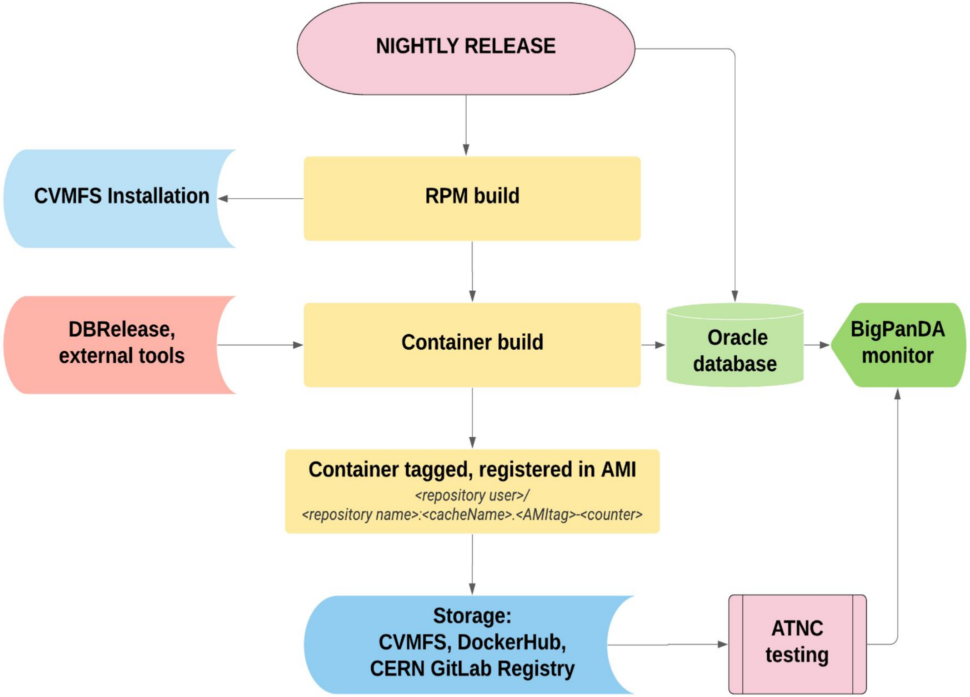
- sites in Germany: LRZ C2PAP, LRZ LMU, MPG Draco
- sites in the Czech Republic: IT4Innovations Barbora and Salomon
- sites in the U.S.: ALCF Theta, NERSC Cori, TACC Frontera

The number of CPU cores (multiplied by their HS06 power) of the simulation jobs (with release 21.0.15) in running status vs time



link to grafana monitoring

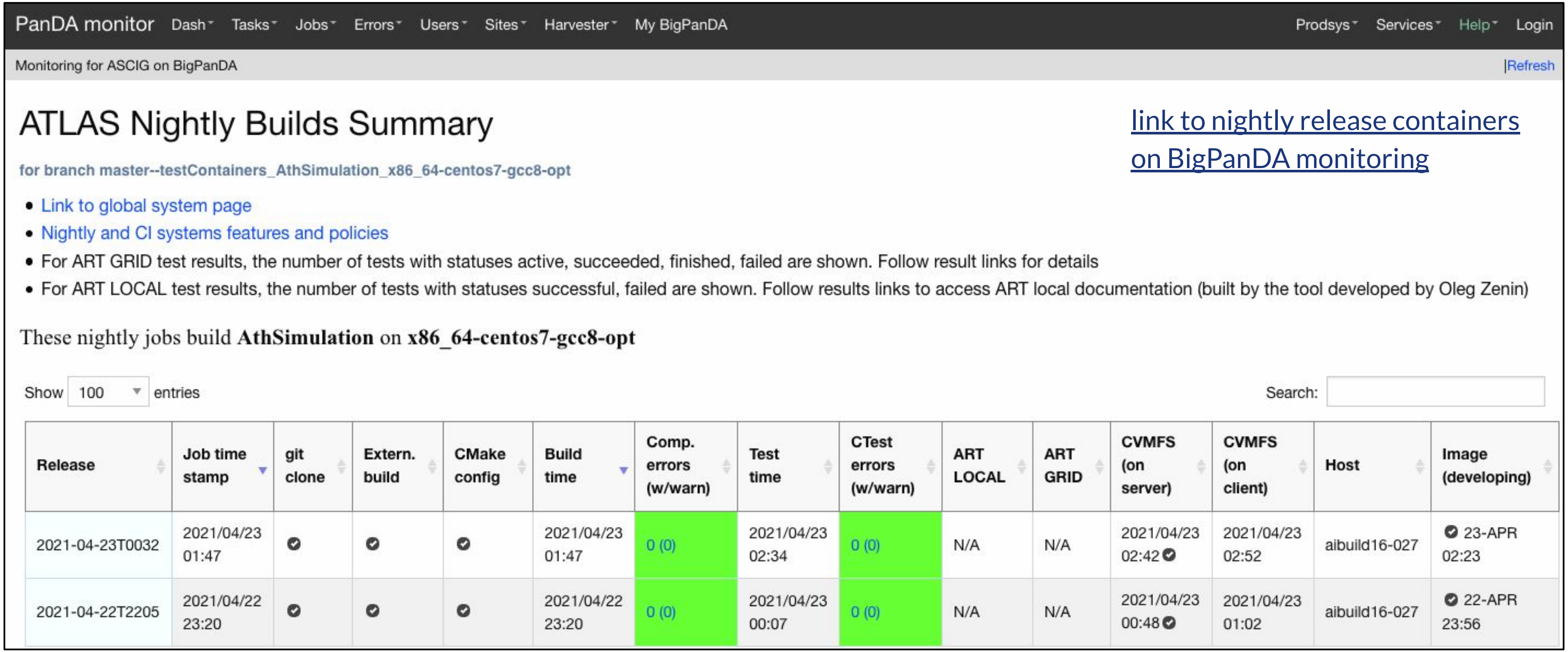
Container Builds in ATLAS Nightly System


NIGHTLY RELEASE
CVMFS Installation
RPM build
DBRelease, external tools
Container build
Oracle database
BigPanDA monitor
Container tagged, registered in AMI
<repository user>/
<repository name>:<cacheName>.<AMItag>-<counter>
Storage: CVMFS, DockerHub, CERN GitLab Registry
ATNC testing

- Container building and registration steps have been added to ATLAS Jenkins-based nightly build system.
- Building nightly release containers allows easy deployment to multiple different operating systems and hardware platforms including those lacking CVMFS access.
- Currently runs once a week on the AthSimulation master branch.
- Nightly release containers kept on CVMFS and Docker Hub for at least 30 days.

Monitoring of Nightly Release Containers

Nightly build jobs of containers are displayed on BigPanDA monitoring system.



Monitoring for ASCIG on BigPanDA

ATLAS Nightly Builds Summary

for branch master--testContainers_AthSimulation_x86_64-centos7-gcc8-opt

- Link to global system page
- Nightly and CI systems features and policies
- For ART GRID test results, the number of tests with statuses active, succeeded, finished, failed are shown. Follow result links for details
- For ART LOCAL test results, the number of tests with statuses successful, failed are shown. Follow results links to access ART local documentation (built by the tool developed by Oleg Zenin)

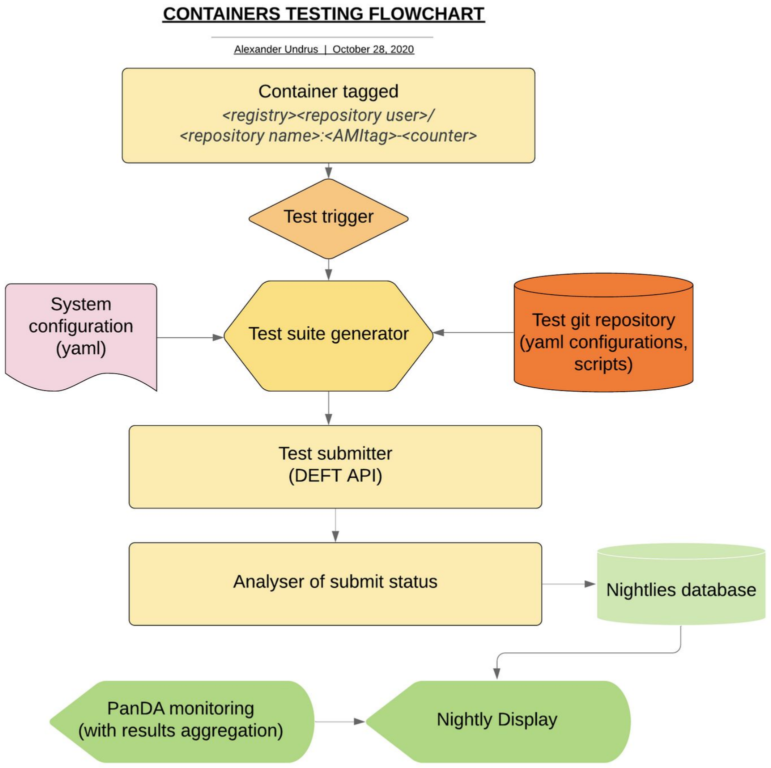
These nightly jobs build **AthSimulation** on **x86_64-centos7-gcc8-opt**

Show 100 entries Search:

Release	Job time stamp	git clone	Extern. build	CMake config	Build time	Comp. errors (w/warn)	Test time	CTest errors (w/warn)	ART LOCAL	ART GRID	CVMFS (on server)	CVMFS (on client)	Host	Image (developing)
2021-04-23T0032	2021/04/23 01:47	✓	✓	✓	2021/04/23 01:47	0 (0)	2021/04/23 02:34	0 (0)	N/A	N/A	2021/04/23 02:42 ✓	2021/04/23 02:52	aibuild16-027	✓ 23-APR 02:23
2021-04-22T2205	2021/04/22 23:20	✓	✓	✓	2021/04/22 23:20	0 (0)	2021/04/23 00:07	0 (0)	N/A	N/A	2021/04/23 00:48 ✓	2021/04/23 01:02	aibuild16-027	✓ 22-APR 23:56

link to nightly release containers on BigPanDA monitoring

Container Testing in ATNC Framework


CONTAINERS TESTING FLOWCHART
Alexander Undrus | October 28, 2020
Container tagged
<registry><repository user>/
<repository name>:<AMItag>-<counter>
Test trigger
System configuration (yaml)
Test suite generator
Test git repository (yaml configurations, scripts)
Test submitter (DEFT API)
Analyser of submit status
Nightlies database
PanDA monitoring (with results aggregation)
Nightly Display

link to ATNC GitLab project

A framework called ATNC (ATLAS Testing Nightly for Containers) has been developed to test the nightly release containers:

- Check that all needed software, externals, databases are included and work together.
- Test the environments setup shipped in the container.
- Check compatibility with ATLAS distributed computing systems and tools.
- Check basic functionalities for which the container is designated (e.g. using one standard physics validation sample to run at ATLAS grid sites for testing the AthSimulation nightly containers).
- Tests can be automatically triggered from the machinery of the nightly build system.
- Results are displayed on BigPanDA monitoring system.
- Framework is extensible to validate stable releases.

Summary and Outlook

- A pipeline has been developed for building software release containers along with their releases in the ATLAS nightly system.
- A testing framework has been put together for the nightly release containers to run validation tests at grid sites.
- This work allowed to efficiently utilize several HPCs in ATLAS production campaigns.
- Containers will continue to be a glue in running ATLAS workflows at opportunistic sites.
    - Effort is underway to add support for the deployment of images to run Monte-Carlo reconstruction.
- In the meantime challenges are upcoming with transition from Docker Hub to GitLab/Harbor registry as well as with transition from the ATLAS Jenkins-based CI system to the GitLab-based CI in 2022.