ATLAS Software Infrastructure

Alexander Undrus

Introductory talk

NPPS group meeting

July 2019

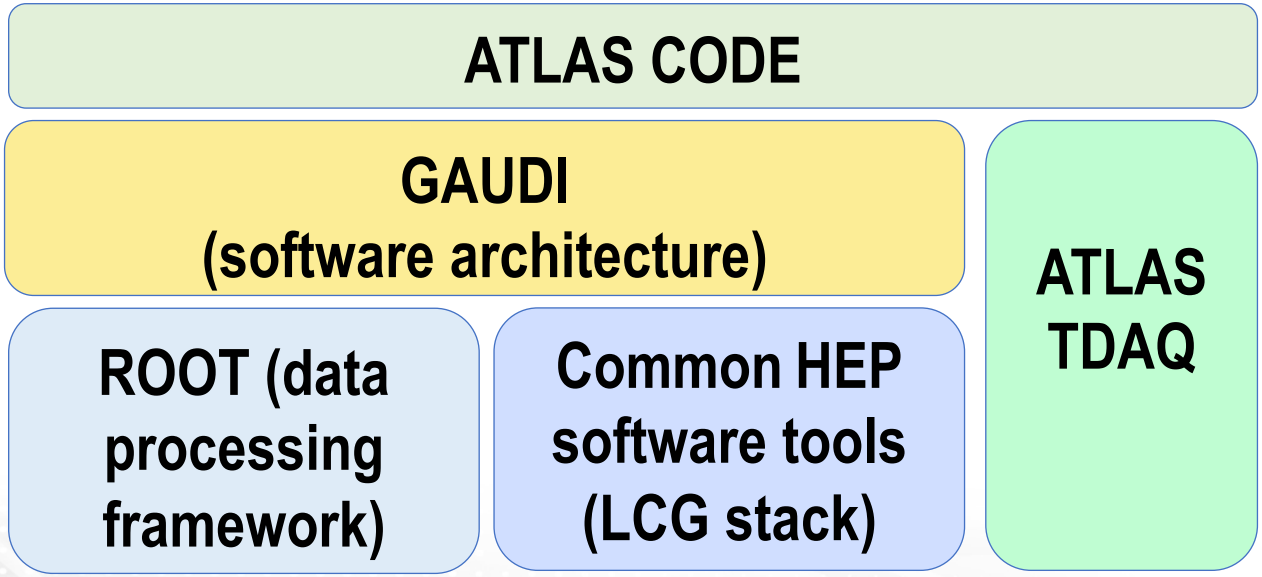
BROOKHAVEN NATIONAL LABORATORY

U.S. DEPARTMENT OF ENERGY

BROOKHAVEN SCIENCE ASSOCIATES

ATLAS Offline Code Base

- All-inclusive “Athena” releases (~ 5 million code lines)
  - Require 240 external packages (mostly supplied by CERN SFT team, ATLAS TDAQ releases, GAUDI architecture framework, generators)
  - Partial releases for Simulation, Analysis available
  - Online software is separate, beyond the scope of this talk



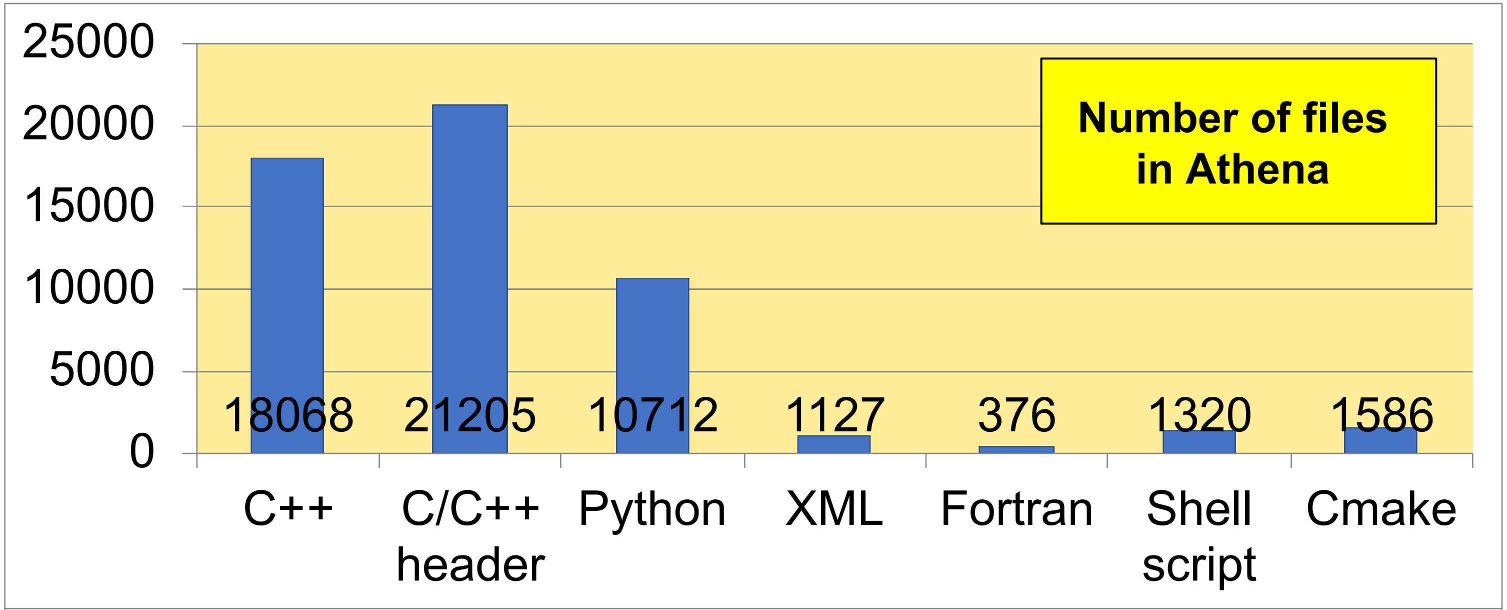
ATLAS CODE

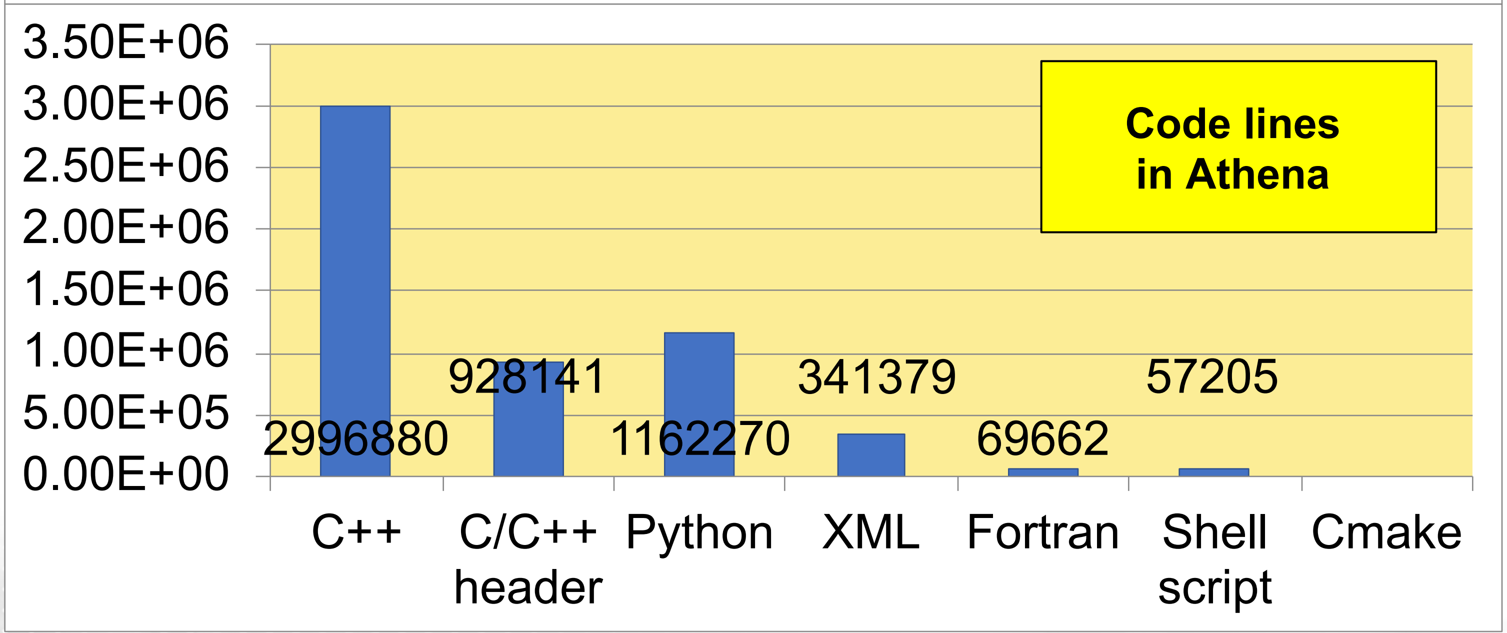
GAUDI (software architecture)

ATLAS TDAQ

ROOT (data processing framework)

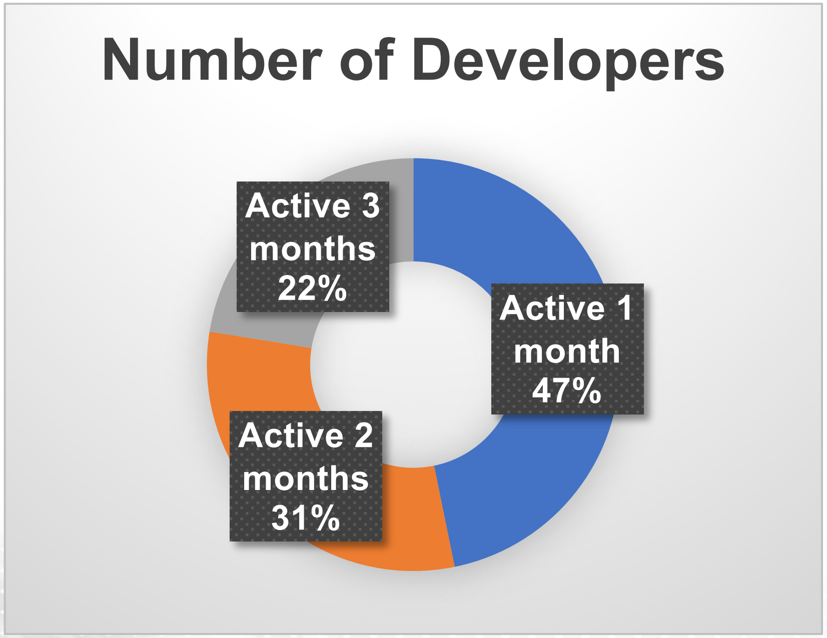
Common HEP software tools (LCG stack)


Number of files in Athena
25000
20000
15000
10000
5000
0
18068
21205
10712
1127
376
1320
1586
C++
C/C++ header
Python
XML
Fortran
Shell script
Cmake


Code lines in Athena
3.50E+06
3.00E+06
2.50E+06
2.00E+06
1.50E+06
1.00E+06
5.00E+05
0.00E+00
928141
341379
57205
2996880
1162270
69662
C++
C/C++ header
Python
XML
Fortran
Shell script
Cmake

ATLAS Developers Community

- Collaboration: 3000 scientists and 1200 students
  - Most of them make contributions to code
- Departures and arrivals are frequent
  - Currently 2 – 3 new developers are granted access to ATLAS Athena project (in GitLab) daily

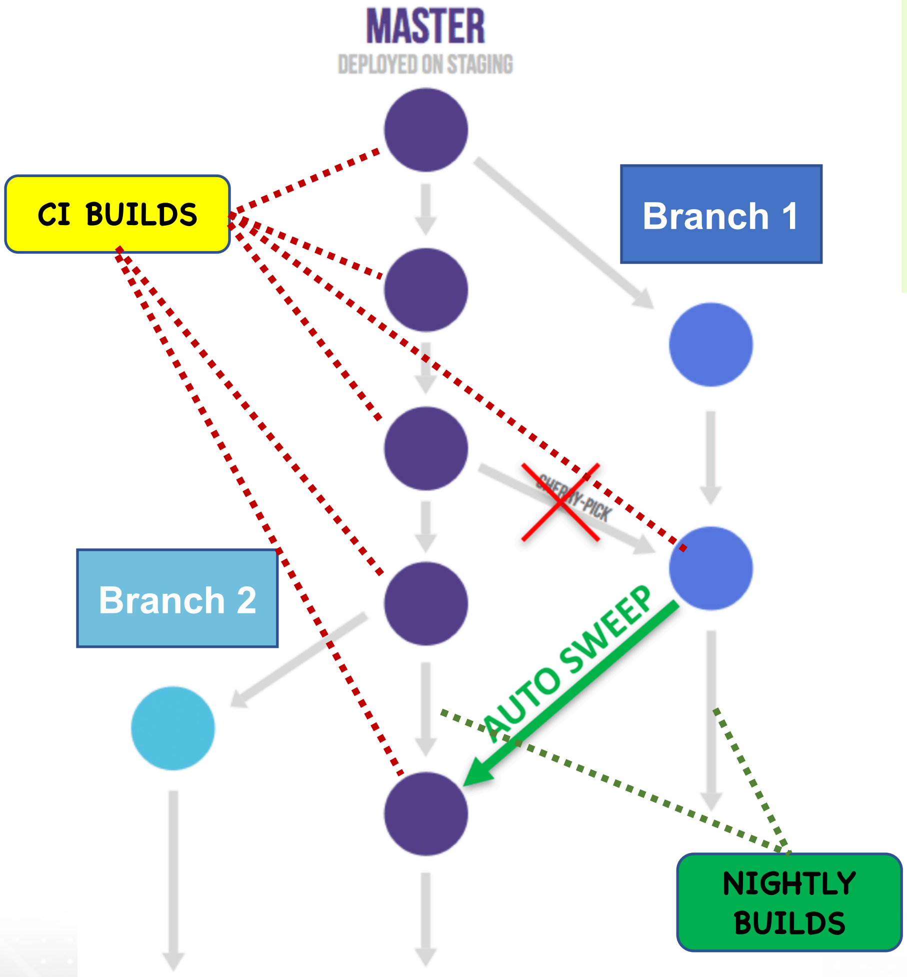


- In 3 monthly periods: *01/21-02/20, 03/21-04/20, 05/21-06/20* **156** developers made **2223** commits to ATLAS Athena repository (merge commits excluded)
- Only **22%** of developers made commits in **all periods**
- **47%** of developers made commits only in **one period**

ATLAS Software Use in Operations

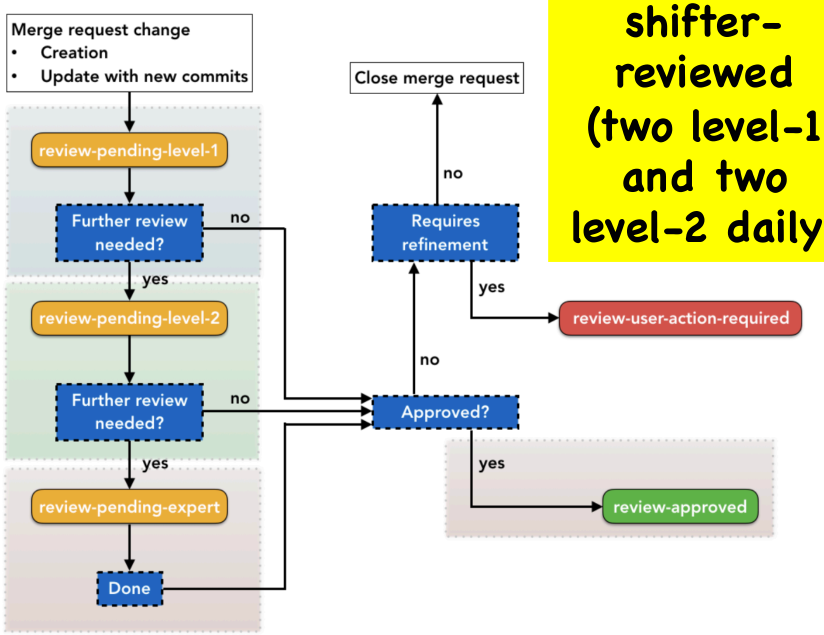
- Collaboration: 3000 scientists and 1200 students
  - Most of them ran ATLAS jobs using offline software
- Global ATLAS operations
  - 30M jobs monthly at > 250 sites
  - 1.4+ Exabytes processed annually
  - 1110 monthly active users

ATLAS Software Development Workflow



ATLAS does not enforce the 'upstream first' policy, but allows for changes to be made directly in release branches. Automated daily 'sweeps' copy those changes into the master branch.

Each MR is shifter-reviewed (two level-1 and two level-2 daily)



US ATLAS Responsibilities

Management of key infrastructure systems

Nightlies and Continuous Integration (CI) (A. Undrus)

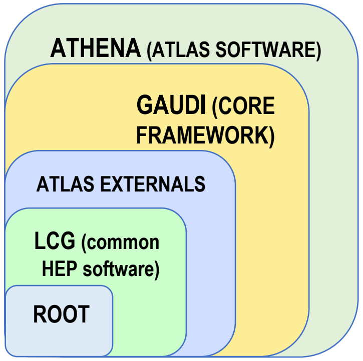
- **Centerpiece** of ATLAS software workflow – Jenkins based build and testing systems interconnected with GitLab
- **Big scale and complexity**
  - **~11000 Athena releases** built in 2019
  - **1530 cores** on build farms
  - **Multiple branches, projects, platforms**
  - svn-, git- based workflow supported
  - **Dynamic monitoring**
  - Continuous systems development as per users request: **71 JIRA tasks** (mostly improvements) were completed in 2019 so far

AtlasSetup build-, run-time environment setup tool (S.Ye)

- **Majority of ATLAS jobs and user sessions start with running AtlasSetup**
- Support of various operating systems, compilers, build tools – used currently or in the past
- Response to users concerns and questions on daily basis

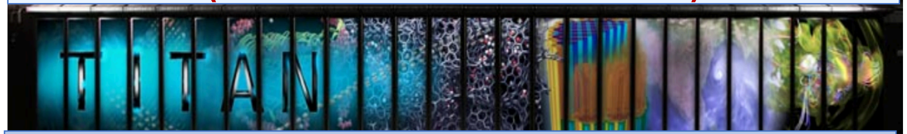
Many interesting projects beyond key responsibilities. Example: ATLAS Comprehensive Software Compilation (ACSC) Project

All-inclusive installation from source code, including generators (Geant4, Pythia…), ROOT, LCG stack



- **Full automation feasible:** code upload via **HTTP** (**no CVMFS**)

Friendly Linux, AMD CPUs
(ATLAS kits binaries work)

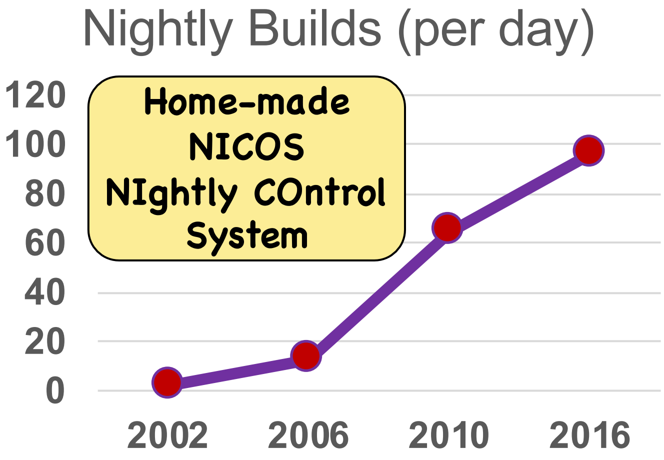


PowerPC, 10X of Titan
IBM CPUs, GNU Linux
(ATLAS kits binaries do **not** work)

RESULTS

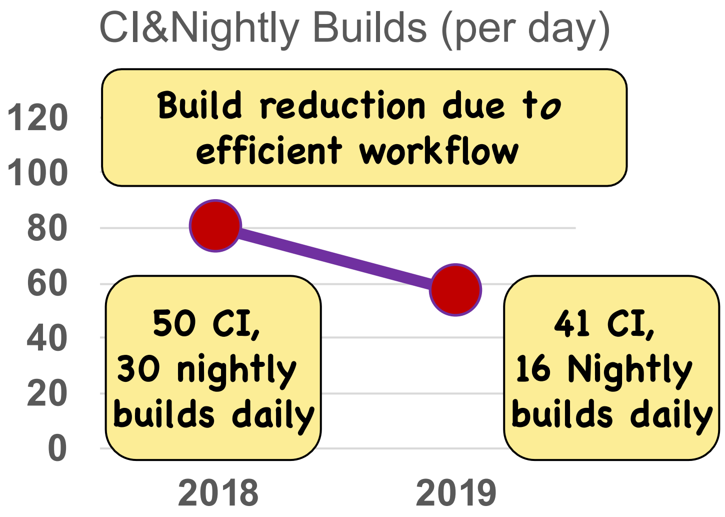
- **Athena release 21.0.31 was installed and tested on Summit**
  - AthSimulation 21.0.34 – Titan, Summit
- **Total compilation time 1 day**
  - **5M ATLAS code lines, 100 externals, 130 generators**
- **Few code adjustments needed (e.g. compiler macro)**

ATLAS Nightly/CI Systems History


Nightly Builds (per day)

Home-made NICOS NIghtly COntrol System

2017 CI transition: Git, Jenkins


CI&Nightly Builds (per day)

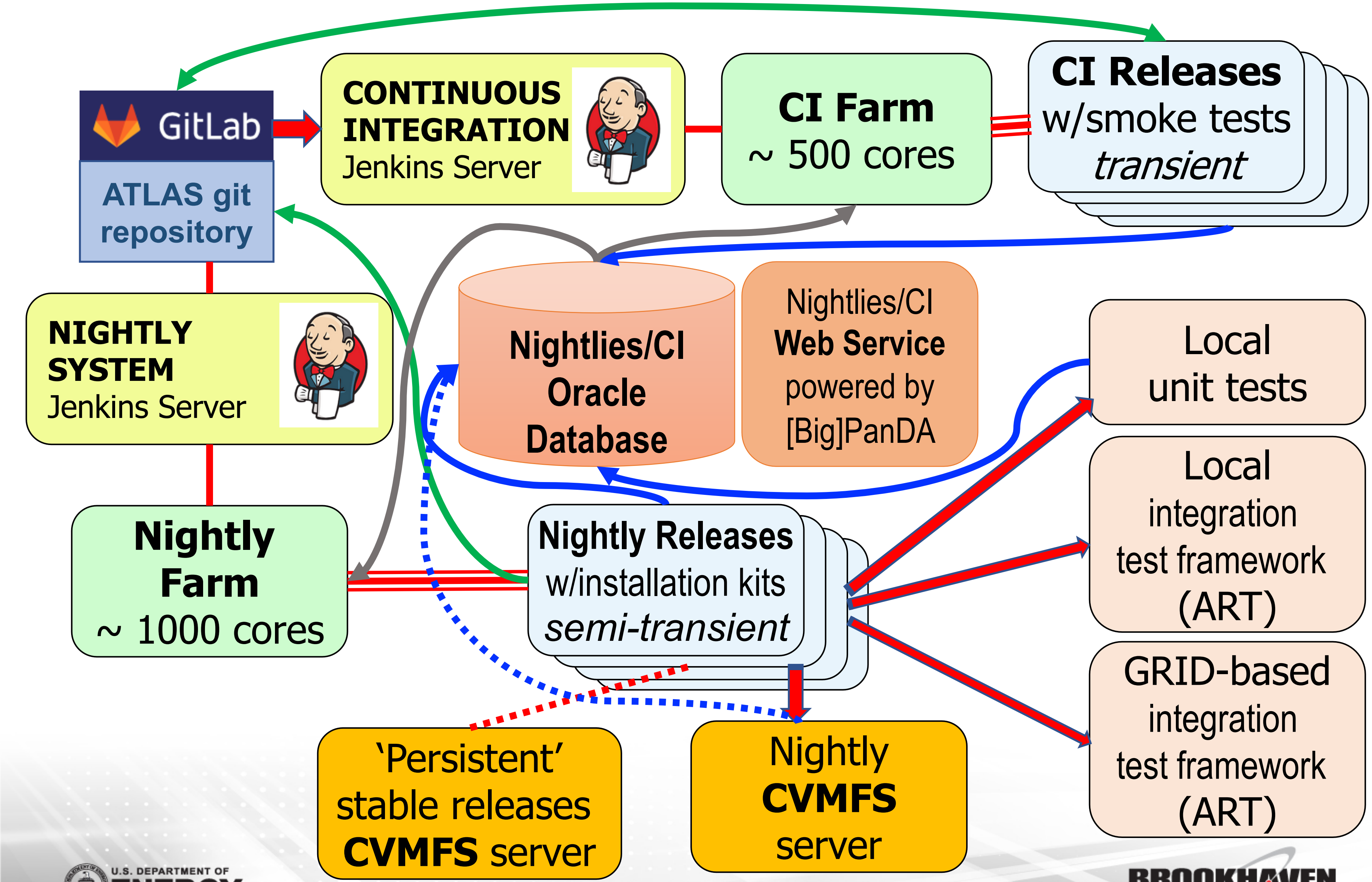
Build reduction due to efficient workflow

50 CI, 30 nightly builds daily

41 CI, 16 Nightly builds daily

- **Today: 23 nightly branches (multiple platforms, projects)**
  - ~ 16 nightly jobs on average day (some branches 'on-demand')
- **CI build for each MR creation/update** (up to 100 daily)
- **Comprehensive testing (unit, local and GRID integration)**
- **Excellent stability**
  - Occasional VM, EOS problems affect << 1% of jobs
  - Hard work: 57 JIRA issues, 44 release installation requests tackled in 6 months of 2019

Jenkins-based Build Systems Details


GitLab
ATLAS git repository
CONTINUOUS INTEGRATION Jenkins Server
CI Farm ~ 500 cores
CI Releases w/smoke tests transient
NIGHTLY SYSTEM Jenkins Server
Nightlies/CI Oracle Database
Nightlies/CI Web Service powered by [Big]PanDA
Local unit tests
Local integration test framework (ART)
Nightly Farm ~ 1000 cores
Nightly Releases w/installation kits semi-transient
GRID-based integration test framework (ART)
'Persistent' stable releases CVMFS server
Nightly CVMFS server

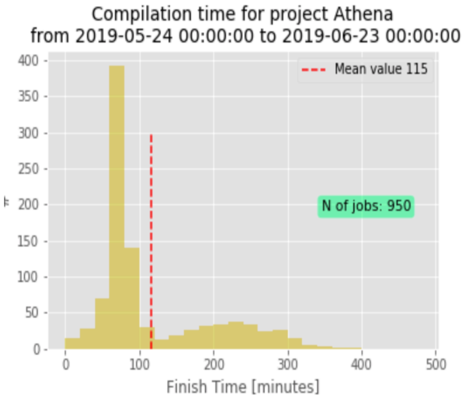
Jenkins-based Build Systems Notes

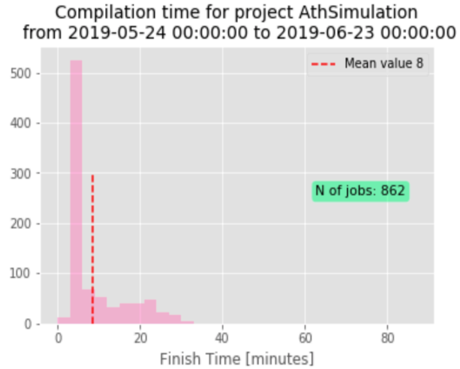
- Build machines are very powerful VMs
  - 16-20 cores, up to 120 GB RAM
  - Fast 0.5 TB SSD (a build job needs > 0.3 TB)
- … and it matters
  - Current release ‘from scratch’ compilation time is 6 hours (faster in CI where most builds are incremental), 10 hours with testing, installations
  - Build time easily doubles on slower machines, oversubscribed machines, conventional disks

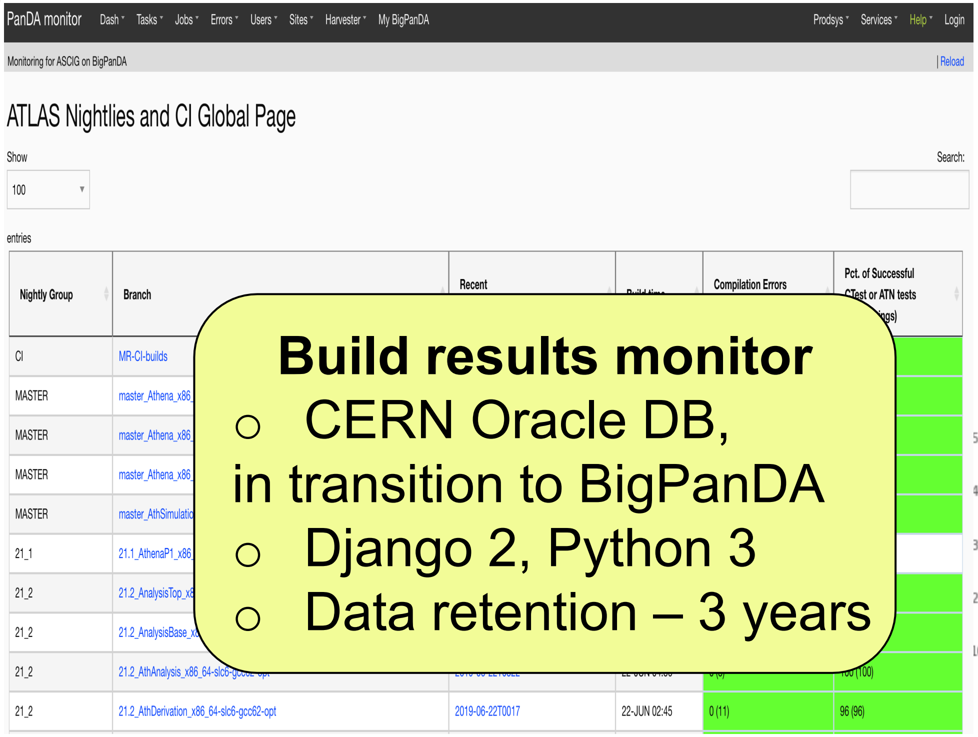
Jenkins Support

- **3 Jenkins instances at CERN compromised in March**
  - Presumably, this was an automated attack with the intention to instantiate crypto-currency mining software on compromised hosts (which didn't succeed).
- Quarter million Jenkins are running around the globe - attractive target for hackers
- Jenkins and its ~50 plugins updates to the latest versions are now performed on our instances ~bi-weekly
  - Require service interruption, tests
- **Plan to keep Jenkins servers behind CERN firewall**
  - SSH tunnels, browser's proxy extensions allow access worldwide

Database-backed Monitoring, Jupyter Analytics







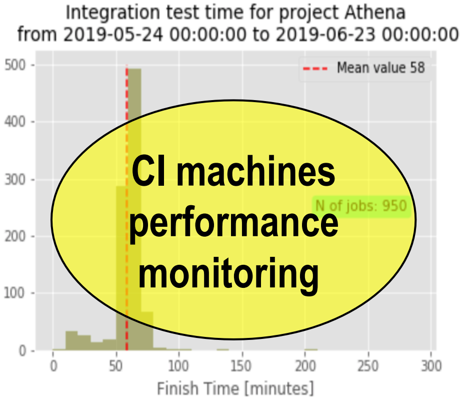
Monitoring for ASCIG on BigPanDA

ATLAS Nightlies and CI Global Page

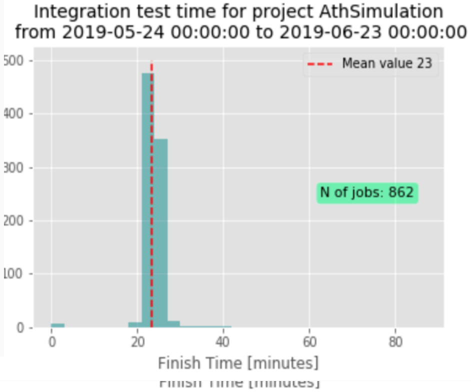
Nightly Group	Branch	Recent	Build time	Compilation Errors	Pct. of Successful CTest or ATN tests ings)
CI	MR-CI-builds				
MASTER	master_Athena_x86_				
MASTER	master_Athena_x86_				
MASTER	master_Athena_x86_				
MASTER	master_AthSimulatio				
21_1	21.1_AthenaP1_x86_				
21_2	21.2_AnalysisTop_x8				
21_2	21.2_AnalysisBase_x8				
21_2	21.2_AthAnalysis_x86_64-slc6-gcc62-opt				100 (100)
21_2	21.2_AthDerivation_x86_64-slc6-gcc62-opt	2019-06-22T0017	22-JUN 02:45	0 (11)	96 (96)

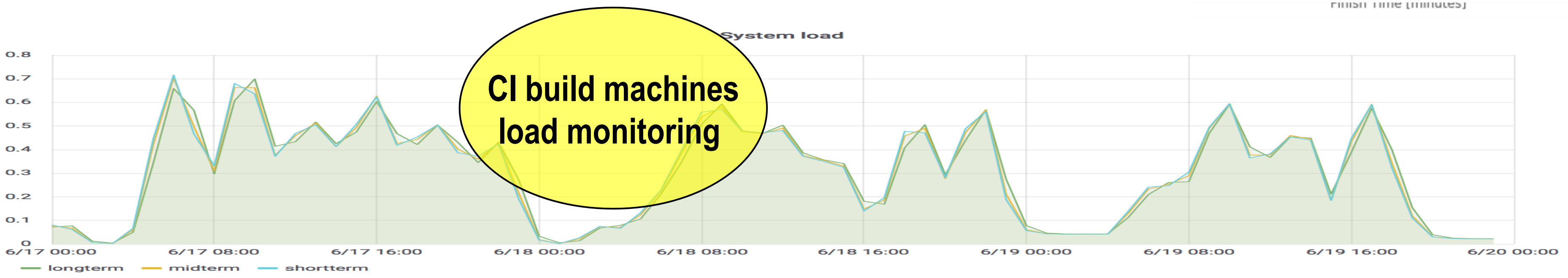
Build results monitor

- ○ CERN Oracle DB, in transition to BigPanDA
- ○ Django 2, Python 3
- ○ Data retention – 3 years



CI machines performance monitoring





CI build machines load monitoring

Plans

- Evaluate GitLab CI (with CERN IT)
  - CERN IT: improvements and new features of GitLab CI makes it easier to implement the ATLAS workflow than before
  - While CERN IT supports Jenkins and GitLab, it does not support the "bridge" between Jenkins and GitLab (”GitLab Jenkins plugin”)
- Monitoring improvements for CI and nightly systems
  - Complete migration to BigPanDA service (joint project with S. Padolski, ATLAS ADC team)
  - More details about build and test results (e.g. ART GRID tests)
  - Enhance tracking of VM performance
- For all systems (CI, Nightlies…):
  - Ensure strong user support, systems reliability and productivity
- Longer term: merge CI and nightly systems (and keep an eye on modern CI tools – Tekton, Cloud Build, Travis…)

Conclusions

- Size and complexity of ATLAS software infrastructure commensurate with grandeur and longevity of the experiment
- State of art CI and Nightlies systems under management of US ATLAS/BNL NPPS serve well in the ATLAS software development workflow
- Plans to keep abreast of modern technologies trends are in place