ATLAS SOFTWARE DEPLOYMENT PLANS

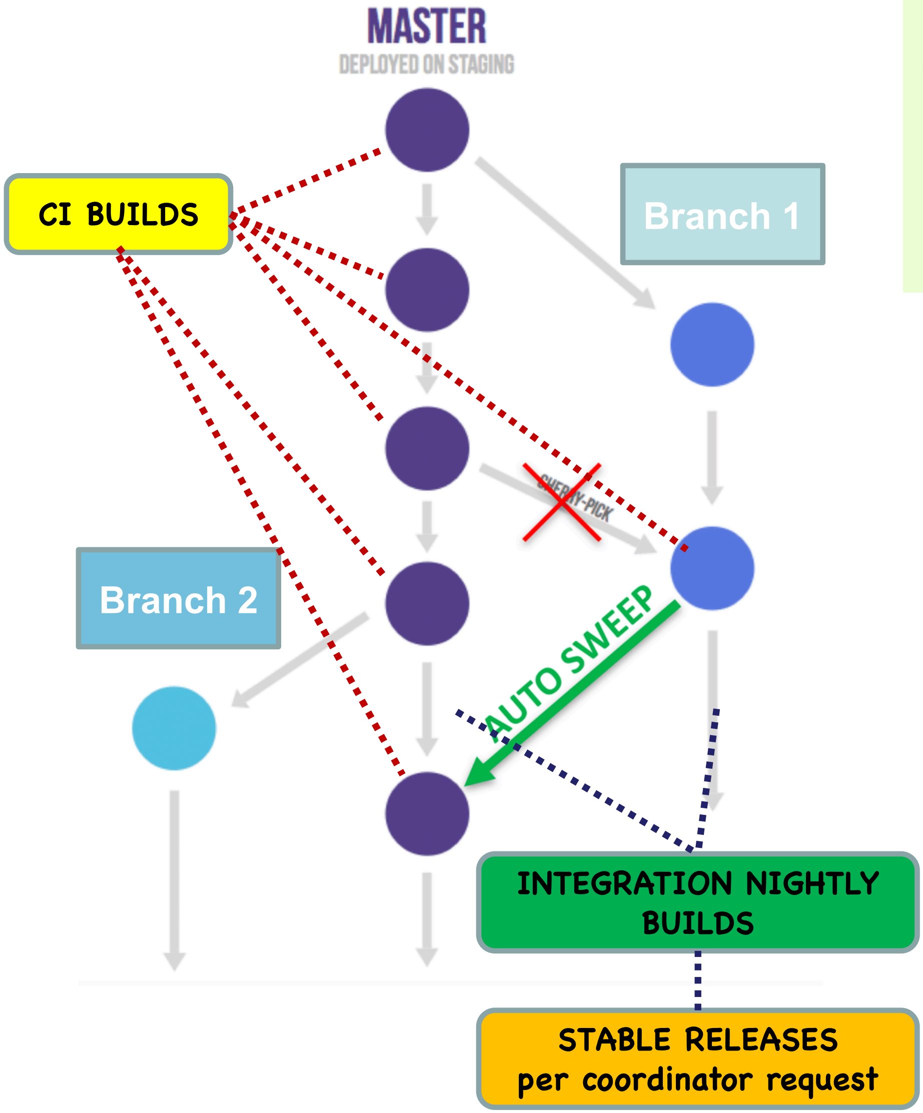
PRE-GDB – SOFTWARE DEPLOYMENT MEETING

MAY 5, 2020

Alexander Undrus (BNL)

ATLAS Software Development Workflow in GitLab

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

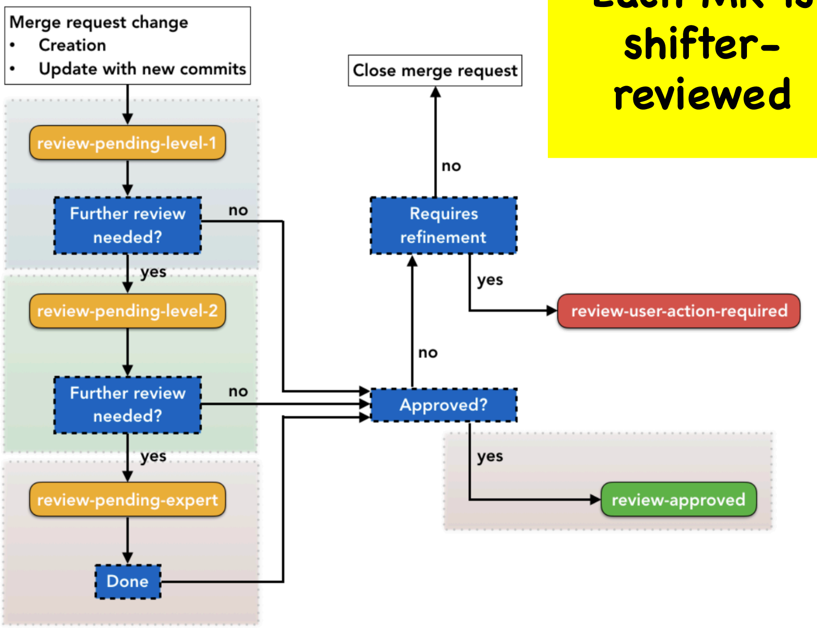


CI BUILDS

INTEGRATION NIGHTLY BUILDS

STABLE RELEASES
per coordinator request

Each MR is shifter-reviewed



Merge request change
- Creation
- Update with new commits

review-pending-level-1

Further review needed? no / yes

review-pending-level-2

Further review needed? no / yes

review-pending-expert

Done

Approved? no / yes

Requires refinement no / yes

Close merge request

review-user-action-required

review-approved

ATLAS Release Building Workflow

Significant hardware resources

- 1200 CPU cores
  (used in systems below)
- >15TB CVMFS
  allocated

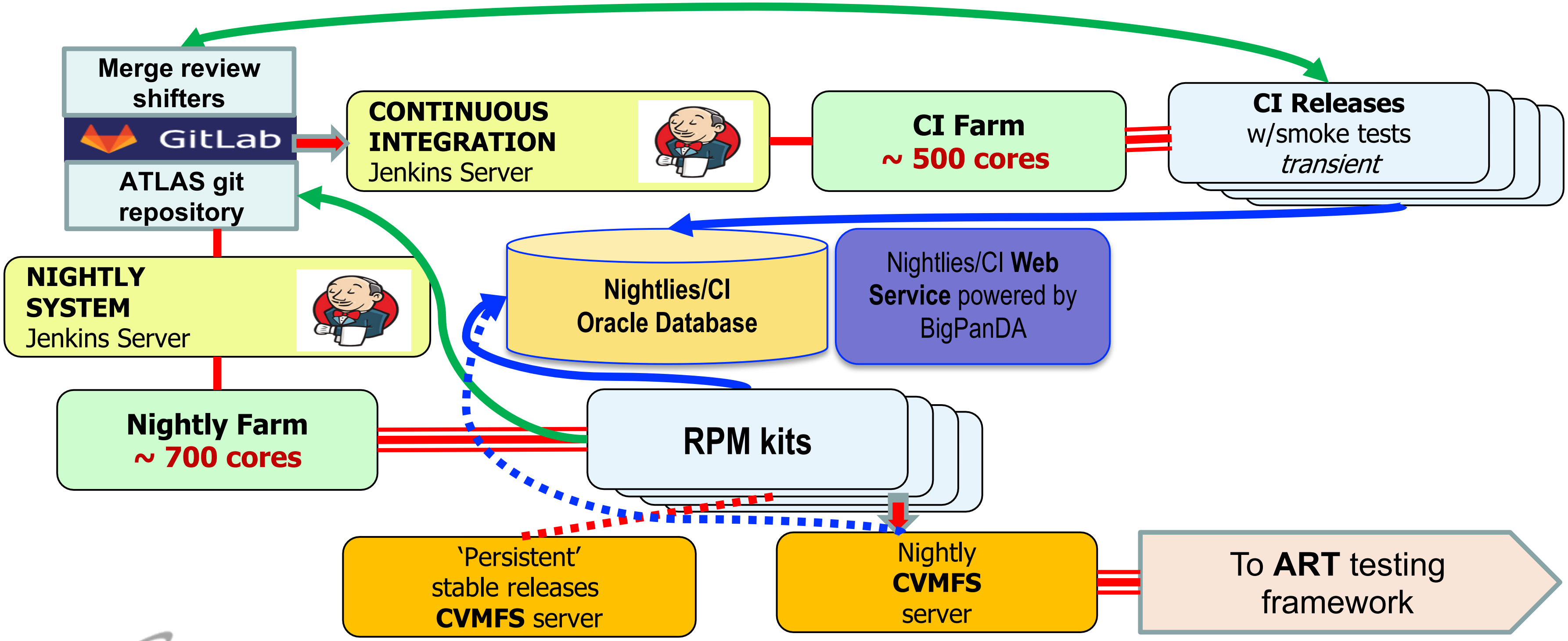
~ 130 stable releases distributed on CVMFS in 2019

- size up to 5 GB
- single platform

Critical importance for ATLAS

3000 scientists, 1200 students

(most of them run ATLAS jobs using offline software)



Merge review shifters

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 BigPanDA

Nightly Farm
~ 700 cores

RPM kits

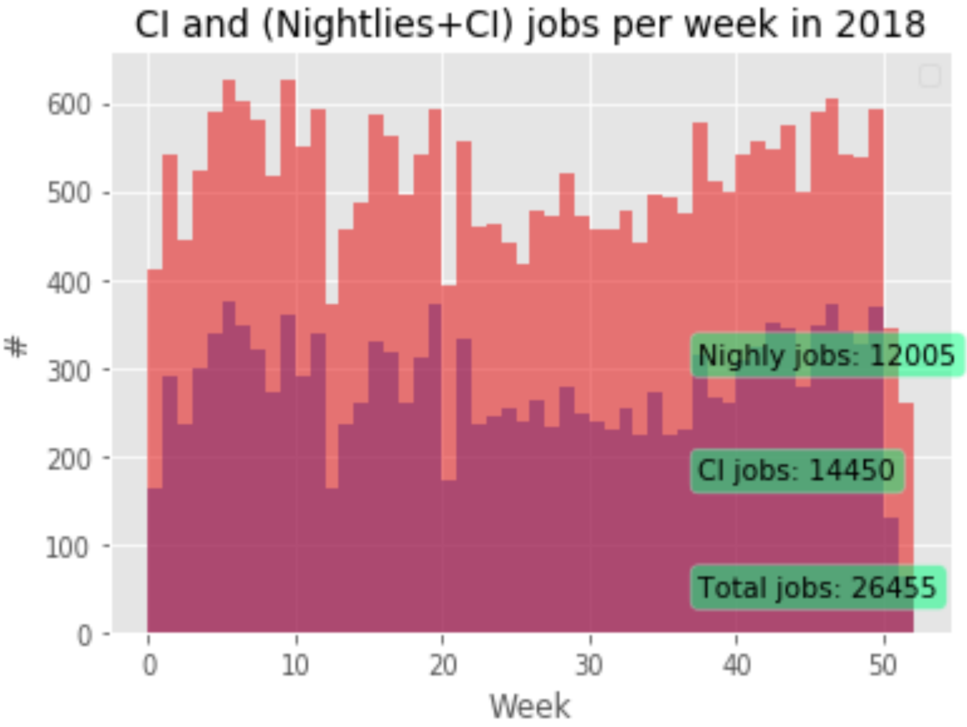
'Persistent'
stable releases
CVMFS server

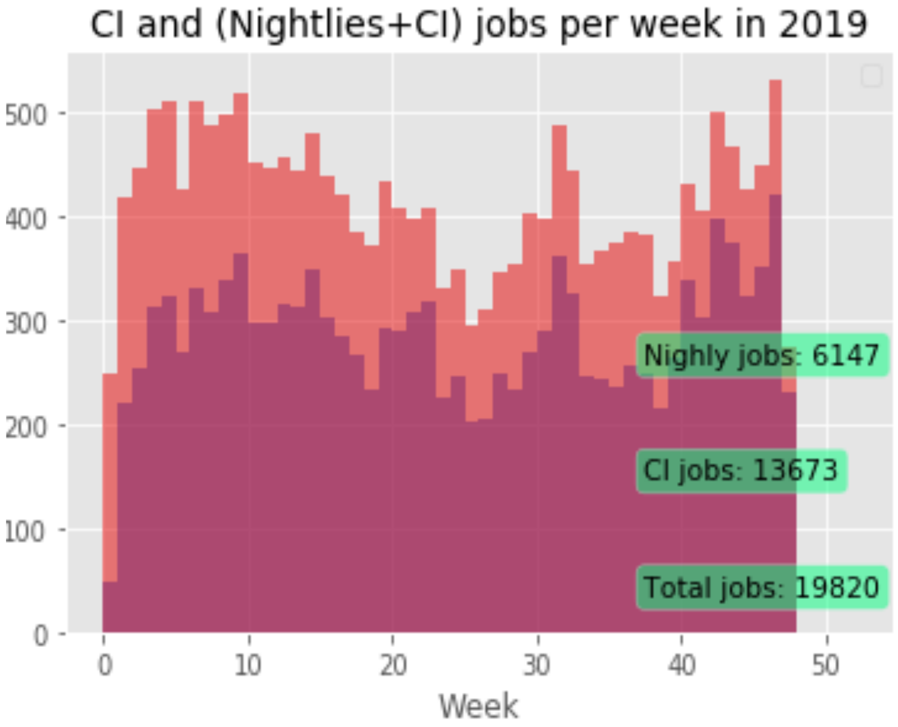
Nightly
CVMFS
server

To **ART** testing framework

Alex Undrus, ATLAS pre-GDB, May 5, 2020

- Each week up to 600 CI, nightly, stable releases are produced for ATLAS developers
- Only nightly and stable releases are deployed


CI and (Nightlies+CI) jobs per week in 2018


CI and (Nightlies+CI) jobs per week in 2019

Current ATLAS Software Deployment

- Single-platform releases
  - Up to 5M code lines, built by CMake
- RPMs are stored on EOS, web accessible
  - Built by CPackRPM generator
- ayum (ATLAS version of yum) is used for installations
  - Enables installations as non-root
- Nightly and stable release are installed on CVMFS
  - Retention period of nightly releases (~ 20 branches) is 30 days
- Containers for some releases available on CVMFS, GitHub
  - "Fat" containers for HPC (emulate a /cvmfs environment, include LCG software stack, Conditions and Trigger data), size ~ 10 GB

BROOKHAVEN
NATIONAL LABORATORY

ATLAS Software Deployment Plans (1/2)

Automated production of the release containers procedure for ATLAS software releases containerization

Multi-domain Task Force

- Integration in the new software development workflow based on GitLab CI
- Registration in ATLAS Metadata Interface ‘ecosystem’
  - Universal naming convention for all ATLAS containers
  - Facilitate containers bookkeeping (retention, etc.)
- “Standalone” containers suitable for HPC and conventional GRID sites
- Automatic CVMFS deployment

ATLAS Software Deployment Plans (2/2)

Streamlining CVMFS installations

- Running CVMFS release managers on the release building machines (testing in progress)
  - Parallel installation of multiple nightly releases
  - Hope for speedy transactions (1 – 2 hours for releases with ~2000 packages are needed)
- CVMFS sync for container deployment *(/cvmfs/unpacked.cern.ch)*
  - Works well for ATLAS statistics software stack containers, deployed multiple times per hour