ATLAS software build system - Part 1

Joint Experiment session - Software builds, CI and more

CERN

6 February 2025

Johannes Elmsheuser (BNL) for the ATLAS ASCIG team

Introduction - setting the scene

List of topics in this talk:

- Cmake
- Externals
- LCG releases
- Projects
- Platforms
- Compilers
- Test builds

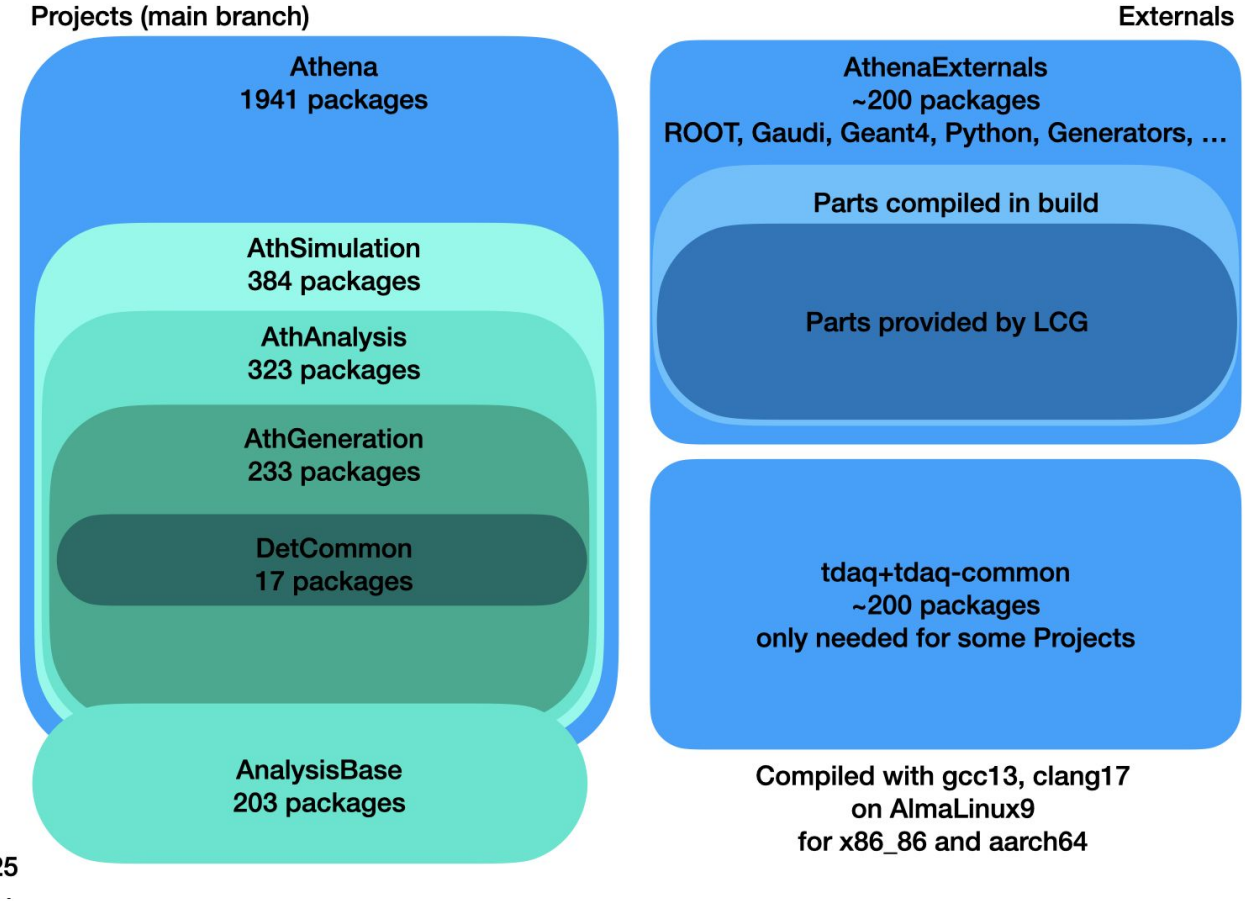
Topics in Alex Undrus' talk:

- Jenkins, Gitlab, CI
- Nightly builds
- Machines
- Numbered releases
- EOS+CVMFS installations
- ART tests

Documentation

- Section 6.1 of Run3 ATLAS software and computing paper
  - https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PAPERS/SOFT-2022-02/
  - https://arxiv.org/pdf/2404.06335
- Athena releases and nightly builds
  - https://atlassoftwaredocs.web.cern.ch/athena/developers/releases/
- ATLAS cmake configuration:
  - https://atlassoftwaredocs.web.cern.ch/athena/developers/cmake/
- ATLAS cmake package:
  - https://gitlab.cern.ch/atlas/atlasexternals/-/tree/main/Build/AtlasCMake
- How to build a release
  - https://atlassoftwaredocs.web.cern.ch/athena/developers/building/
- ATLAS offline software in gitlab
  - https://gitlab.cern.ch/atlas/athena and https://gitlab.cern.ch/atlas/atlasexternals
- Hep_OSlibs meta-package
  - https://gitlab.cern.ch/linuxsupport/rpms/HEP_OSlibs/-/blob/el9/README-el9.md

ATLAS software schematic overview



27 Jan 2025

3 major blocks built by different teams:

- **Projects:** ATLAS offline software
- **Externals:** CERN EP/SFT through LCG layers and ATLAS offline software
- **TDAQ**: ATLAS trigger/DAQ

Operating system, git and platforms

- Operating system
  - Run3: AlmaLinux9 + Hep_OSlibs meta-package (link)
    - Bare metal machines (64 core AMD EPYC) provided by CERN IT for nightlies and CI
  - Run2: CentOS7 nightlies once per 10 days built in container
- Offline software repository and CI
  - Open source and hosted at https://gitlab.cern.ch/atlas/athena/ and https://gitlab.cern.ch/atlas/atlasexternals
  - Using jenkins CI for athena MRs and gitlab CI for atlasexternals MRs
  - 2 gitlab branches of athena under active development:
    - main (Evgen, derivations, Analysis), 24.0 (Tier0 reconstruction, MC simulation, HLT at Point1)
- Platforms
  - x86_64-v2 and aarch64 (Arm v8) (N.B. about 2% of Grid jobs with x86_64-v2)
  - gcc 13.1 (Run3 production), gcc14.2, clang17 with cuda 12.4.1 and clang19 with cuda 12.8
  - Run2 legacy builds with gcc11 and gcc8
- Containers
  - Created for AnalysisBase/AthAnalysis every week
  - Created on demand for other releases occasionally

Projects

- Full Athena build of ~2000 packages supports (theoretically) most of the production workflows
  - Athena, AthSimulation, AthAnalysis/AnalysisBase, AthGeneration, DetCommon
  - Projects with dedicated use cases developed years ago to make builds a bit more light-weight and save build time (when build machines were not yet so powerful)
  - Uses LCG_107 as basis and tdaq/tdaq-common (Athena, DetCommon) for externals
  - Can be used via CVMFS
- AnalysisBase is an exception:
  - Full stand-alone ROOT based release (no LCG and tdaq dependencies)
  - Work horse for end-user analysis
  - Can be used via CVMFS and also with containers on e.g. a laptop

LCG releases and AtlasExternals

- Software stack provided by CERN EP/SFT with
  - Content overview: https://lcginfo.cern.ch
  - More details in “2025 CERN EP/SFT program of work” overview talk at link
  - Provides compilers and consistent software stacks: 800+ external packages
  - Major LCG versions in sync with major/bugfix versions of ROOT
  - ATLAS requests LCG layers usually with minor version updates of MC generators and a few other external packages every few weeks
- AtlasExternals
  - Dedicated builds of externals ATLAS prefers to control version/patches:
    - Geant4 (10.6 used for Run3)
    - Gaudi
    - Several python packages, onnxruntime, gdb and more
    - ROOT for AnalysisBase

CMake and Packages

- Historically ATLAS offline code is organized in ~2000 packages, i.e. translates into ~2000 subdirectories with dedicated CMakeLists.txt files that represent e.g. a particular reconstruction algorithm+tools
- CMake:
  - Used to build libraries and executables together with python configuration and data files
  - A dedicated set of CMake macros was developed to ease usage of package structure vs. libraries (link and link)
  - EOS and CVMFS are used for large data files for CI tests
  - User can develop code and build small parts against a full pre-build nightly (located on CVMFS)
  - CTest is used for a total of ~2400 unit tests
  - CPack is used to create RPMs used for nightly installation on CVMFS
  - If deemed good a nightly can be declared a numbered stable release, will separately installed on CVMFS with ‘unlimited lifetime and can be directly used in production
  - N.B. nighlies can also be used in production but will unless pinned disappear after 30 days)

CMakeLists.txt examples for ATLAS package

main / athena / Control / AthenaExamples / AthExHelloWorld / **CMakeLists.txt** — Find file | Blame | History | Permalink

Cleanup AthExHelloWorld
Tadej Novak authored 8 Mar 2023 at 18:56 — Verified — 7905c150

Code owners Assign users and groups as approvers for specific file changes. Learn more. — Manage branch rules

CMakeLists.txt 1.26 KiB — Edit | Lock | Replace | Delete

```
# Copyright (C) 2002-2023 CERN for the benefit of the ATLAS collaboration

# Declare the package name:
atlas_subdir( AthExHelloWorld )

# Component(s) in the package:
atlas_add_component( AthExHelloWorld
                     src/*.cxx
                     src/components/*.cxx
                     LINK_LIBRARIES GaudiKernel AthenaBaseComps )

# Install files from the package:
atlas_install_joboptions( share/*.py )
atlas_install_python_modules( python/HelloWorldConfig.py
                              POST_BUILD_CMD ${ATLAS_FLAKE8} )

# Test(s) in the package:
atlas_add_test( AthExHelloWorld
                ENVIRONMENT THREADS=0
                SCRIPT test/test_AthExHelloWorld.sh )

atlas_add_test( AthExHelloWorldMT_1
                ENVIRONMENT THREADS=1
                SCRIPT test/test_AthExHelloWorld.sh )

atlas_add_test( AthExHelloWorldMT_2
                ENVIRONMENT THREADS=2
                SCRIPT test/test_AthExHelloWorld.sh
                LOG_IGNORE_PATTERN "AthenaHiveEventLoopMgr.* processing event|^HelloWorld .*(INFO|WARNING A WARNIN
)

atlas_add_test( AthExHelloWorldCA
                SCRIPT python -m AthExHelloWorld.HelloWorldConfig
                POST_EXEC_SCRIPT nopost.sh )
```

main / athena / Reconstruction / Jet / JetRec / CMakeLists.txt — Find file | Blame | History | Permalink

JetRec: thread-checker cleanup
Frank Winklmeier authored 23 Jan 2025 at 08:29 — e81bbbda

Code owners Assign users and groups as approvers for specific file changes. Learn more. — Manage branch rules

CMakeLists.txt 2.99 KiB — Edit | Lock | Replace | Delete

```
# Copyright (C) 2002-2025 CERN for the benefit of the ATLAS collaboration

# Declare the package name:
atlas_subdir( JetRec )

# Extra dependencies, based on the environment:
set( extra_libs )
if( NOT GENERATIONBASE )
   list( APPEND extra_libs xAODPFlow )
   if( NOT XAOD_ANALYSIS )
      list( APPEND extra_libs AthenaMonitoringKernelLib )
   endif()
   if( NOT XAOD_STANDALONE )
      list( APPEND extra_libs StoreGateLib )
   endif()
endif()

# External dependencies:
find_package( FastJet COMPONENTS fastjetplugins fastjettools siscone siscone_spherical )
find_package( FastJetContrib COMPONENTS VariableR RecursiveTools )
find_package( GTest )

# Component(s) in the package:
atlas_add_library( JetRecLib
  JetRec/*.h Root/*.h Root/*.cxx src/*.cxx
  PUBLIC_HEADERS JetRec
  INCLUDE_DIRS ${FASTJET_INCLUDE_DIRS} ${FASTJETCONTRIB_INCLUDE_DIRS}
  LINK_LIBRARIES ${FASTJET_LIBRARIES} ${FASTJETCONTRIB_LIBRARIES} AsgDataHandlesLib AnaAlgorithmLib AsgTools  AthL
  PRIVATE_LINK_LIBRARIES CxxUtils xAODTracking )

if( NOT XAOD_STANDALONE )
  atlas_add_component( JetRec
    src/components/*.cxx
    LINK_LIBRARIES ${FASTJET_LIBRARIES} AthContainers AthenaBaseComps GaudiKernel JetEDM JetInterface JetRecLib St
endif()
```

Left: https://gitlab.cern.ch/atlas/athena/-/blob/main/Control/AthenaExamples/AthExHelloWorld/CMakeLists.txt
Right: https://gitlab.cern.ch/atlas/athena/-/blob/main/Reconstruction/Jet/JetRec/CMakeLists.txt

Nightly build flavours

- Building nightlies from 24.0 and main gitlab branches
  - Athena, AthSimulation, AthGeneration, AthAnalysis, AnalysisBase, DetCommon
  - x86_64 and aarch64 with gcc13.1
  - Athena nightly with clang17, clang19 and gcc14
- Development nightlies
  - dev3LCG (ROOT head), dev4LCG (currently v6-34-00-patches)
  - Test new versions of ROOT, external packages, cmake, cuda etc.
- Several dedicated experimental special nightlies
  - ACTS, archflag (x86_64-v3), lto, gccchecker, HepMC2, Gaudi
- Legacy nightlies
  - Run2 CentOS7 builds in containers

Summary and Conclusions

- Provided a brief overview of Part 1 of the ATLAS offline software build system and dependencies
- More details in the documentation links provided on slide 3

Acknowledgments - Thanks to Dario Barberis, Chris Lee, Attila Krasznahorkay, Alexander Undrus, Oana Vickey Boeriu, Brinick Simmons, Frank Winklmeier, Shuwei Ye, Asoka De Silva, John Chapman, James Catmore