



Modernizing the CMS software stack

Mircho Rodozov
On behalf of the CMS collaboration

Outline

- Introduction
- CMS Software stack
 - How big/complex is our stack
- Modernizing our stack
 - Python2-3 support
 - Code formatting
- CI/CD improvements & “exotic” experiments
 - Heterogeneous computing
 - Managing CI for 120+ github repositories
 - CI resources utilization



Introduction

➤ We are:

- Core software @ CMS
- Build releases, Integration builds (IBs), externals, test & deploy them
- Continuous integration / delivery (CI/CD) system
- Maintaining CI/CD infrastructure

CMS Software stack size

- Size of CMS Offline Software (CMSSW)
 - Over 4.5 M lines of code
 - 3.2 M C/C++ 1.4 M Python, 275k Fortran
 - Packages - 1300+
 - Multiple versions (10+ release flavors)

- Dependency on external packages
 - 500+, from source
 - custom build system (cmsBuild)

- Contributions
 - 200+ commits/week
 - 160+ github pull requests (PRs)
 - 800+ contributors

- 200+ GitHub repositories for CMSSW and externals

Software Distribution rate

- Release builds
 - Every two weeks
 - via github issues

Build CMSSW_11_0_0_pre11 #28287

Open fabiocos opened this issue 3 days ago · 44 comments

fabiocos commented 2 days ago Member Author +@ ...

+1

Labels

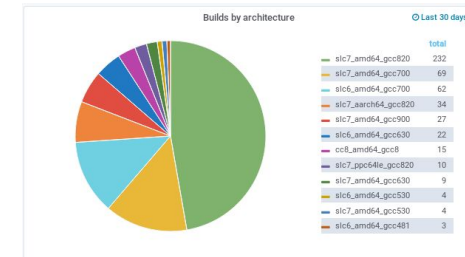
- release-announced
- release-build-request
- release-notes-requested
- slc7_aarch64_gcc820-upload-ok
- slc7_amd64_gcc820-installation-ok
- slc7_amd64_gcc900-installation-ok

- Integration Builds
 - ~450/month

CMSSW_11_0_X_2019-10-29-2300

IB Tag Comparison Baseline DOM Tests HLT Validation Valgrind Code complexity metrics Flaw finder IgProf Static Analyzer SA thread unsafe SA failures 1 SA thread unsafe EventSetup products Header consistency FWLite GPU unit tests Python3

	DEFAULT				ROOT6_X	ROOT618_X	DEVEL_X	CLANG_X
	slc7 amd64 gcc820	slc7 aarch64 gcc820	cc8 amd64 gcc8	slc7 amd64 gcc900	slc7 ppc64le gcc820	slc7 amd64 gcc820	slc7 amd64 gcc820	slc7 amd64 gcc820
	Full Build	Full Build	Full Build	Full Build	Full Build	Full Build	Full Build	Full Build
Builds	⊕	⊕	⊕	5	10	⊕	⊕	⊕
Unit Tests	⊕	4	40	1	14	1	⊕	2
RelVals	3082	35	39	6	32	3082	3082	3082
Other Tests	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
Q/A	🔍	🔍	🔍	🔍	🔍	🔍	🔍	🔍



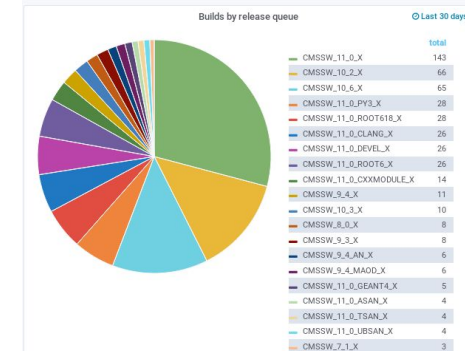
- Complexity

CMSSW IB page CMSSW release: 11_0_X 10_6_X 10_3_X Older: ▾

Flavor: DEFAULT UBSAN_X TSAN_X ROOT6_X ROOT618_X PY3_X GEANT4_X DEVEL_X CLANG_X ASAN_X

OS: slc7 cc8 CPU: amd64 aarch64 ppc64le Compiler: gcc820 gcc8 gcc900

nextIB



- Extensive validation of IBs
 - 4000+ tests





CI setup



submit PR

DQM: Reduce TH1 Usage #28342

Open schneiml wants to merge 24 commits into `cms-sw:master` from `schneiml:dqm-reduce-th1-usage`

cmsbuild added `code-checks-pending` `comparison-pending` `dqm-pending` `geometry-pending` `orp-pending` `pending-signatures` `reconstruction-pending` `simulation-pending` `tests-pending` labels

8 days ago

A new Pull Request was created by @kpedro88 (Kevin Pedro) for master.

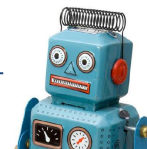
It involves the following packages:

- FastSimulation/SimplifiedGeometryPropagator
- RecoTracker/Configuration
- RecoTracker/DeDx

+code-checks

Logs: <https://cmsstdt.cern.ch/SDT/code-checks/cms-sw-PR-28254/12414>

- This PR adds an extra 24KB to repository



please test

+1

Tested at: `3ffdf28`



+1



+1



+1

cmsbuild added `reconstruction-approved` and removed `reconstruction-pending` labels

cmsbuild added `orp-approved` and removed `orp-pending` labels 2 days ago

cmsbuild added `fastsim-approved` and removed `fastsim-pending` labels 4 days ago

- `code-checks-approved`
- `comparison-available`
- `fastsim-approved`
- `fully-signed`
- `orp-approved`
- `reconstruction-approved`
- `tests-approved`





Code format

- Clang format and clang tidy
 - Formatted 3.2 M lines
 - Collision with open PRs avoided
 - Style used: Google
 - Rules enforced via PR testing

➤ Automated transition

Running code-format for pdmv #27357

Merged cmsbuild merged 2 commits into master from code-format-pdmv-198474 on Jun 27

Conversation 12 Commits 2 Checks 0 Files changed 61



cmsbuild commented on Jun 26

Member + 😊 ...

Applying code-format for CMSSW category pdmv. See the build logs here <https://cmsdt.cern.ch/jenkins/job/GitHub-refactor-cmssw-module/439/console>

cms-bot has successfully run the following:

- scram build code-checks-all
- scram build code-format-all
- scram build

Clang format

```

33 31+ class BeamSplash : public edm::EDFilter {
34 32 public:
35 - explicit BeamSplash(const edm::ParameterSet &);
33 + explicit BeamSplash(const edm::ParameterSet &);
36 34 ~BeamSplash() override;
37 -
35 +
38 36 private:
39 - bool filter(edm::Event &, const edm::EventSetup&) override;
40 -
37 + bool filter(edm::Event &, const edm::EventSetup&) override;
38 +

```

Clang tidy

```

78 - if ( ERecoHitCollection_.label() != "" && ERecoHitCollection_.instance() != "" )
78 + if ( !ERecoHitCollection_.label().empty() && !ERecoHitCollection_.instance().empty() )
79 79 {
80 80     iEvent.getByLabel( ERecoHitCollection_, pERecoHits);
81 81     if ( pERecoHits.isValid() )
81 ✖ @@ -88,7 +88,7 @@ bool BeamSplash::filter( edm::Event& iEvent, const edm::EventSetup& iSetup)
88 88     }
89 89     }
90 90
91 - if ( EERecoHitCollection_.label() != "" && EERecoHitCollection_.instance() != "" )
91 + if ( !EERecoHitCollection_.label().empty() && !EERecoHitCollection_.instance().empty() )
92 92 {
93 93     iEvent.getByLabel( EERecoHitCollection_, pEERecoHits);
94 94
94 ✖ @@ -102,7 +102,7 @@ bool BeamSplash::filter( edm::Event& iEvent, const edm::EventSetup& iSetup)

```

Modernizing python

- 1.2 M lines converted to be py3 compatible
- Migration completed
 - Made it compatible while running py2
 - Works with both
- Mostly a manual job
 - Limited number of cases
 - Some conversion tools exist (futures, autopep)



```

9 + from builtins import int
10
11 ROOT.gSystem.Load("libFWCoreFWLite")
12 ROOT.FWLiteEnabler.enable()
@@ -512,7 +512,7 @@ def to (self, entryIndex):
512     """Jumps to event entryIndex"""
513     if self._veryFirstTime:
514         self._createFWLiteEvent()
- return self._event.to ( Long(entryIndex) )
515 + return self._event.to ( int(entryIndex) )

```

```

7 + if sys.version_info[0]>2:
8 +     import _pickle as cPickle
9 + else:
10 +     import cPickle

```




Python packaging upgrade

- Integrated 200+ python externals
 - Build from sources using pip
- Simplified format
 - one line update
 - py2 and py3 recipes unified
- Customization for the new packages
 - a package can have patch
 - build only for py2 or py3 if required
 - conditionally build
 - python version
 - architecture

Before

```
Source: https://github.com/matplotlib/matplotlib/archive/v%{realversion}.tar.gz
Requires: py2-pytz py2-numpy py2-python-dateutil zlib libpng freetype py2-pyparsing py2-six
BuildRequires: py2-pip

%prep
%setup -n matplotlib-%{realversion}
cat >> setup.cfg <<- EOF
[directories]
basedirlist = ${FREETYPE_ROOT}:${LIBPNG_ROOT}:${ZLIB_ROOT}:${PY2_NUMPY_ROOT}:${PY2_PYTZ_ROOT}:${PY2_SIX}
[gui_support]
gtk = False
gtkagg = False
tkagg = True
wxagg = False
macosx = False
EOF
mkdir no-pkg-config
(echo '#!/bin/sh'; echo 'exit 1') > no-pkg-config/pkg-config
chmod +x no-pkg-config/pkg-config

%install
export CPLUS_INCLUDE_PATH=${FREETYPE_ROOT}/include/freetype2:${LIBPNG_ROOT}/include/libpng16
export MPLCONFIGDIR=$PWD/no-pkg-config
export PATH=$PWD/no-pkg-config:$PATH \
export CPLUS_INCLUDE_PATH=${FREETYPE_ROOT}/include/freetype2:${LIBPNG_ROOT}/include/libpng16
export PYTHONUSERBASE=%i
pip install . --user
pip3 install . --user
```

After

```
numpy==1.15.1 ; python_version<'3.0' ; platform_machine=='aarch64'
numpy==1.15.1 ; python_version>'3.0' ; platform_machine=='aarch64'
oamap==0.12.4
onnx==1.6.0
ordereddict==1.1
packaging==19.2
matplotlib==2.2.4 ; python_version<'3.0'
matplotlib==3.1.1 ; python_version>'3.0'
```

Requires: py2-cycler py2-kiwisolver py2-pyparsing py2-python-dateutil
Requires: zlib libpng freetype

Validating externals

➤ Integration tests for externals


- Does it integrates (build + link against the IB)
- Does it run

Backport OpenBLAS fixes [10_6_X] ✓
comparison-available
externals-pending
orp-pending
pending-signatures
tests-approved
 #5291 opened 9 days ago by kpedro88

Backport OpenBLAS fixes [10_2_X] ●
comparison-available
externals-pending
orp-pending
pending-signatures
tests-rejected
 #5289 opened 9 days ago by kpedro88

➤ Physics validation

- Comparison with reference


cmsbuild commented 36 minutes ago

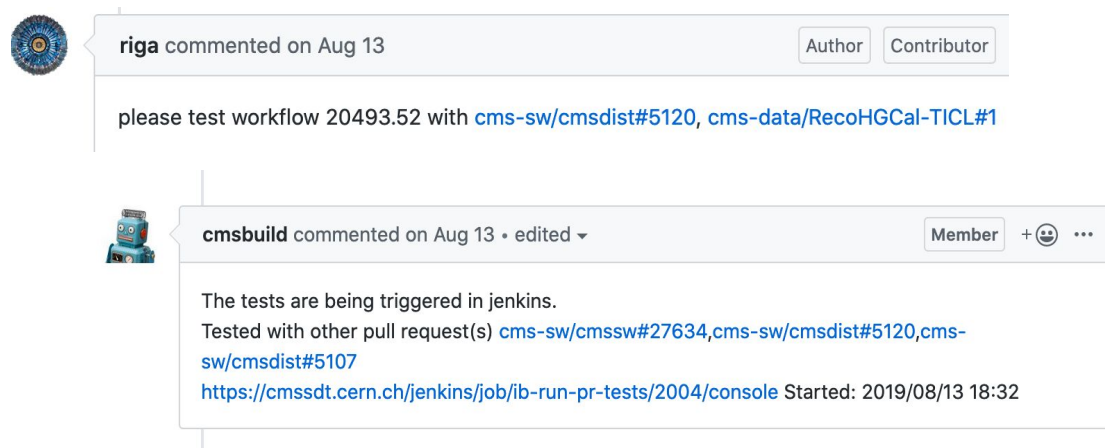
Comparison is ready
<https://cmssdt.cern.ch/SDT/jenkins-artifacts/pull-request-integration/PR-de759a/3262/summary.html>

Comparison Summary:

- No significant changes to the logs found
- Reco comparison results: 0 differences found in the comparisons
- DQMHistTests: Total files compared: 33
- DQMHistTests: Total histograms compared: 3212324
- DQMHistTests: Total failures: 1
- DQMHistTests: Total nulls: 0
- DQMHistTests: Total successes: 3211989
- DQMHistTests: Total skipped: 334
- DQMHistTests: Total Missing objects: 0
- DQMHistSizes: Histogram memory added: 0.0 KiB (32 files compared)
- Checked 137 log files, 14 edm output root files, 33 DQM output files

Testing multiple PRs

- Testing 200+ different repos (CMSSW, externals, data)
- Testing multiple PRs from different repos at once
- Convenient for testing pull requests that need external update



The screenshot shows two GitHub comments. The first comment is from user 'riga', dated August 13, with 'Author' and 'Contributor' roles. The text says: 'please test workflow 20493.52 with [cms-sw/cmsdist#5120](#), [cms-data/RecoHGCal-TICL#1](#)'. The second comment is from 'cmsbuild', dated August 13, with 'Member' role and a '+ 😊 ...' icon. The text says: 'The tests are being triggered in jenkins. Tested with other pull request(s) [cms-sw/cmssw#27634](#), [cms-sw/cmsdist#5120](#), [cms-sw/cmsdist#5107](#) <https://cmssdt.cern.ch/jenkins/job/ib-run-pr-tests/2004/console> Started: 2019/08/13 18:32'.

CMSSW on heterogeneous resources

➤ ARM

- Cavium ThunderX 96 cores
- Cavium ThunderX2 128 cores

➤ PowePC

- Power 8
- 1 Wistron 256 core

➤ GPUs

- Shared Nvidia Tesla K20X
- HTCondor GPUs resources
 - as Jenkins nodes
 - released on job completion
 - Thanks to IT-CM for adding a short queue to access these nodes for us !

slc7 aarch64 gcc820 Full Build	cc8 amd64 gcc8 Full Build	slc7 amd64 gcc900 Full Build	slc7 ppc64le gcc820 Full Build
🟢	🟢	5	10
4	40	1	14
35	39	6	32
🟢	🟢	🟢	🟢
🔍	🔍	🔍	🔍

CI resources utilization

- We run our own scheduler for CI validations
- Heuristics based, 20 to 25% faster
- <https://tinyurl.com/yywh2py7>



Conclusions

- CMS improves it's code base continuously
- Keep improving our CI/CD system
- Embraced modern code policies
- Migrated python
- Successfully running CI and tests on non Intel architectures
- Put some effort for better utilization of the CI resources