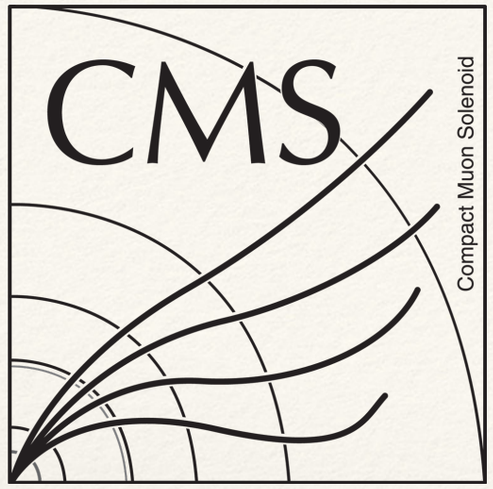


CMS Software Deployment

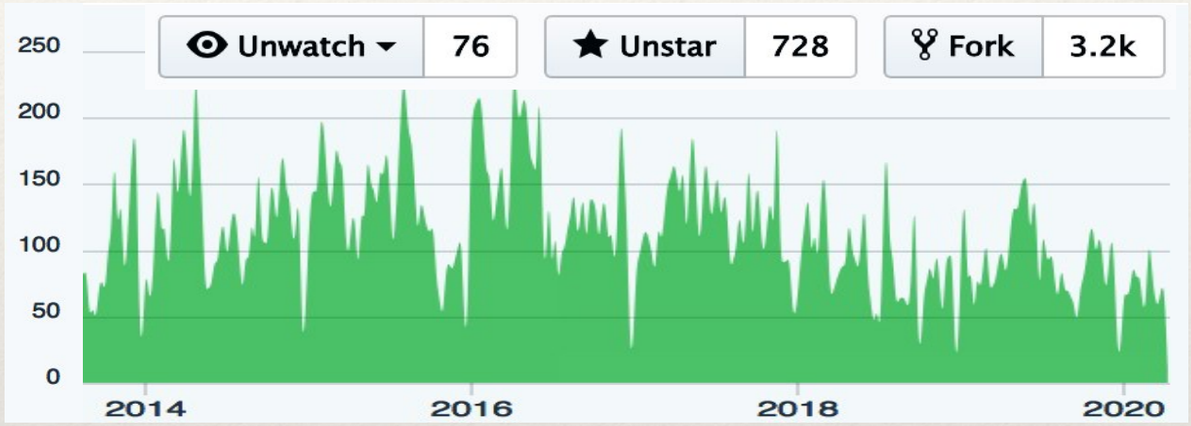
Malik Shahzad MUZAFFAR

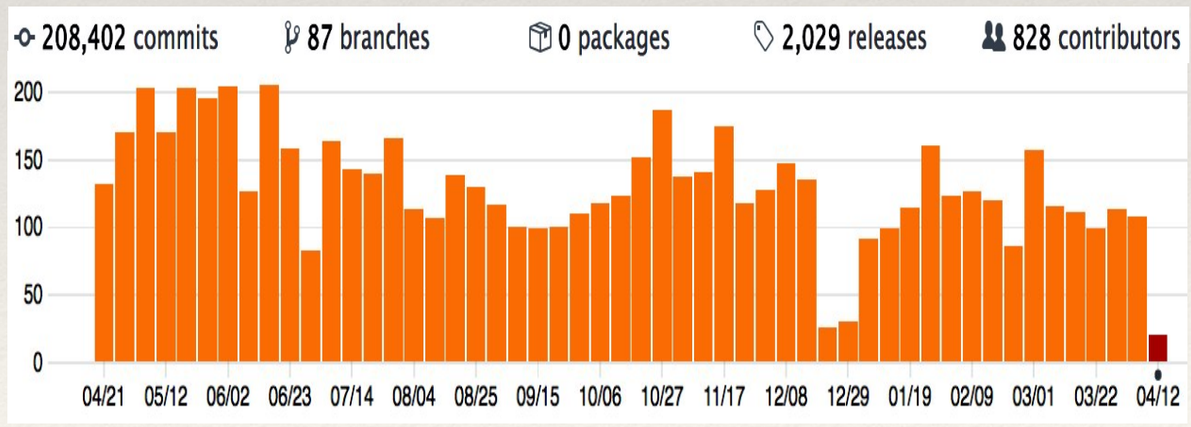
CERN

pre-GDB Software Deployment Meeting
5th May 2020

CMS Offline Software

- CMS Offline Software (CMSSW) has a large code base
  - Open source
    - Hosted on Github since 2013
  - Over **5M lines** of code
    - **65% C++, %5 Fortran**
    - **30% Python**
- **1250+** Packages
  - **3100+** Binary products
    - libs, plugins, executables
- **100+** contributors/month
- **1K+** commits/month

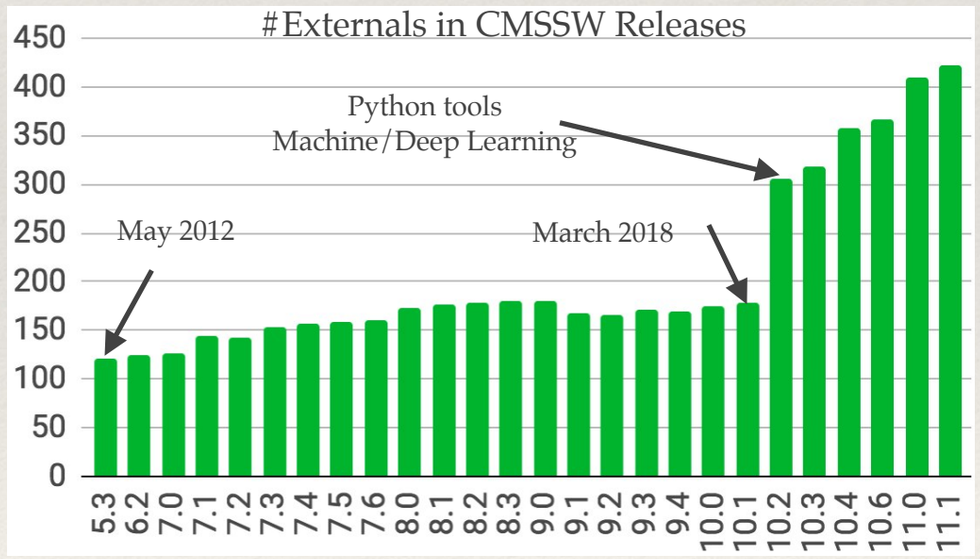




CMSSW Releases

- 9 active release cycles
  - CMSSW **5.3.X - 11.1.X**
  - Various data taking and TDRs

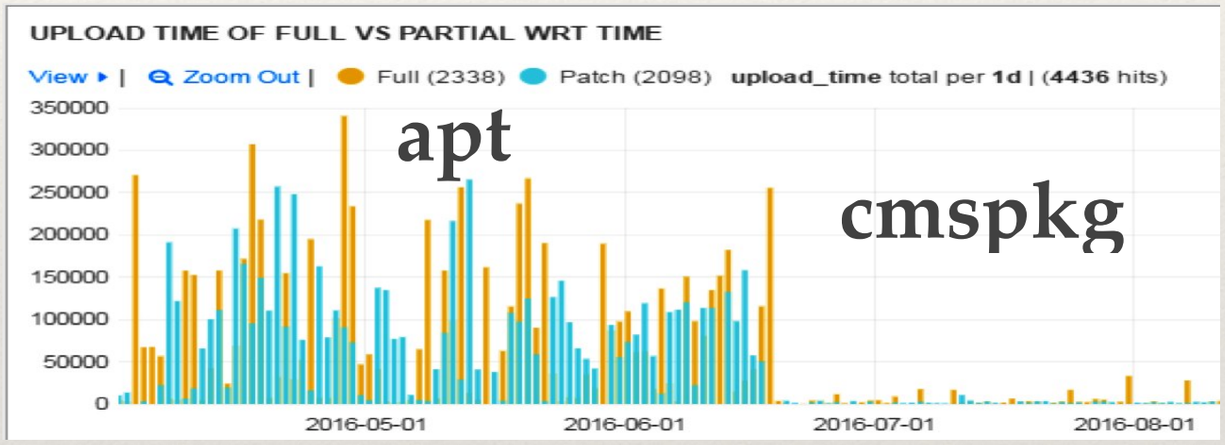
- Built and deployed for **15** active architectures
  - **CC6, CC7, CC8**
  - **Intel, ARM, Power**
  - **GCC 4.8 - 9.3**
- Over **400** open source external tools



- CI, Integration and release build Infrastructure: **530 Cores** (OpenStack)

CMSSW Distribution

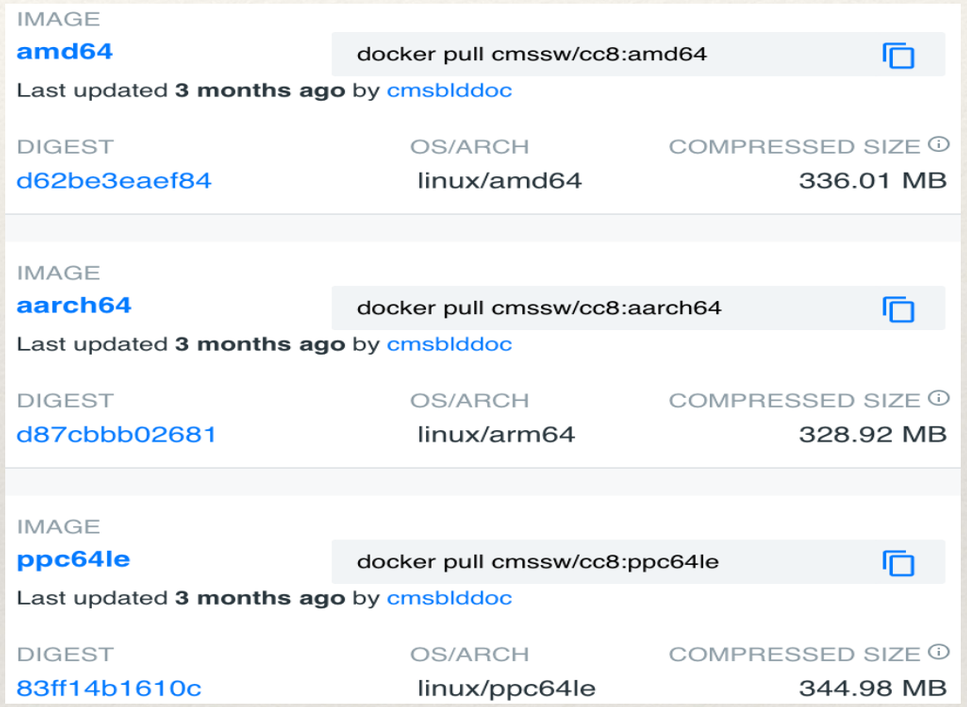
- CMSSW and its dependencies are built and distributed in form of relocatable RPMs
  - Packaging: **cmsBuild**
  - Deployment: **cmspkg**
    - **APT** was used till mid 2016



- **150K** packages for over **60** architectures
  - slc5/ia32 - cc8/amd64, ARM , Power etc.

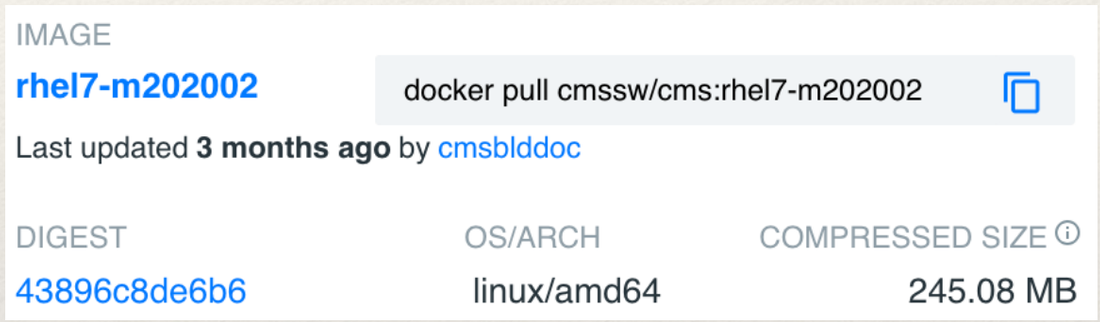
Software Platform: Development

- CMS offline SW environment heavily make use of containers
  - CI, Integration Builds (nightlies), Release build
  - **CC6, CC7, CC8**
    - Based on CentOS images
    - O(150) packages , no HEP_OSlibs
  - **Intel, ARM, Power**
- Automatically built, tested and uploaded
  - DockerHUB
- Available on **unpacked.cern.ch** CVMFS repository
  - **cmssw-cc*N*** script to get CMS environment using singularity
    - Works for all supported architectures: Intel, ARM, Power
  - **Takes few hours to get new image deployed**



Software Platform: GRID

- CMSSW production and analysis jobs run under singularity
  - Light weight containers based on **opensciencegrid/osg-wn:3.4**
    - **EL6, EL7**
    - **Intel** only
    - Extra packages needed by CMSSW (no HEP_OSlibs)



- Automatically built and uploaded to DockerHUB when base image is changed
- Available on **singularity.opensciencegrid.org** CVMFS repository
- We are looking in to supporting users' containers for analysis jobs

Software Platform: HPC

- Each HPC site is different
  - Negotiate deployment of required services
    - Singularity and CVMFS to setup environment and access CMSSW releases
- Prepare appropriate application images
  - Docker containers with pre-installed CMSSW release and conditions data
    - 10-15GB containers

Software Platform: Outreach

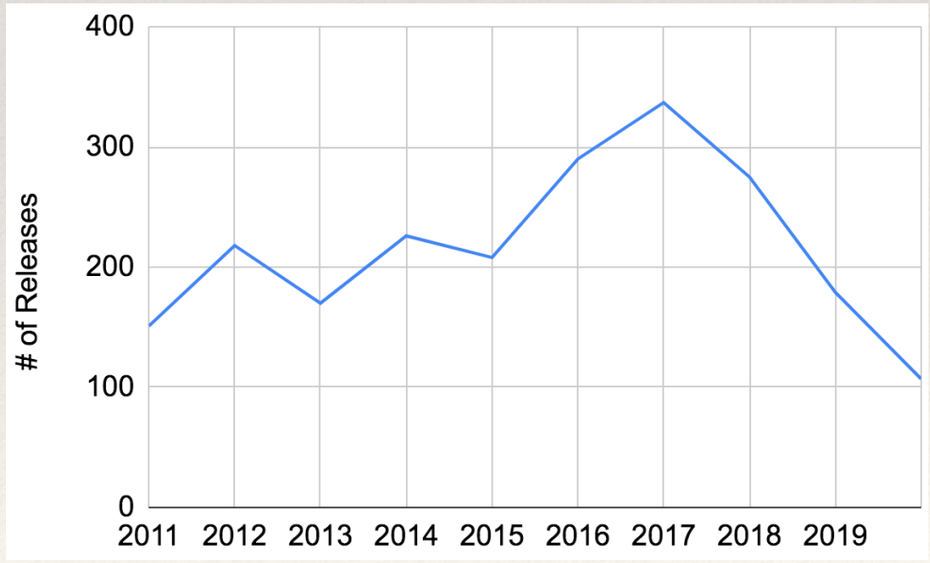
- CMS uses Virtual Machine Images for CMS Open-data
  - Based on CernVM4
    - 40GB virtual disk (20GB CVMFS cache)
- CMSSW and conditions data is accessed through CVMFS
  - **cms.cern.ch**
  - **cms-opendata-conddb.cern.ch**
- XRootD is used to access CMS Open-data

CMSSW Release Deployment

- All CMSSW releases are available via CVMFS **cms.cern.ch**
  - **2100+** full/patch releases
  - **7.2TB**
  - **6** Stratum 1's
    - Low publish frequency, no garbage collection
- Automatically deployed via Jenkins CI after the build
- Who uses these releases? Every one
  - CMS Grid production and user analysis jobs
  - Outreach and HPC
  - CMSSW developers
    - Local installation also possible: 30GB

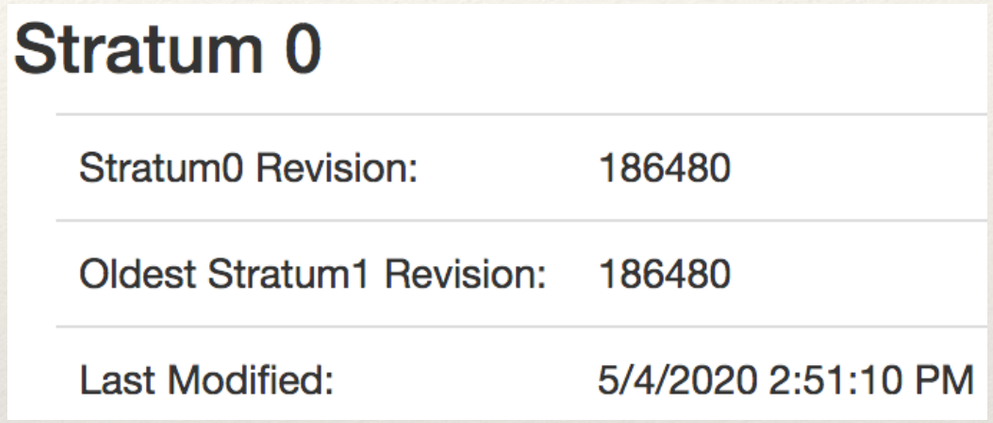
Stratum 0

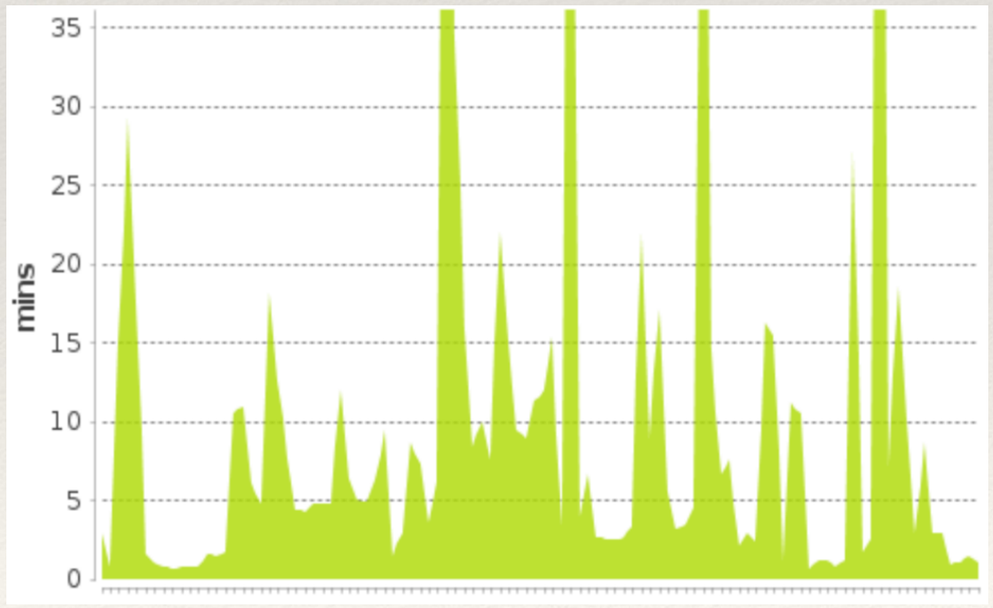
Stratum0 Revision:	92525
Oldest Stratum1 Revision:	92522
Last Modified:	5/4/2020 3:43:19 PM



Integration Builds Deployment

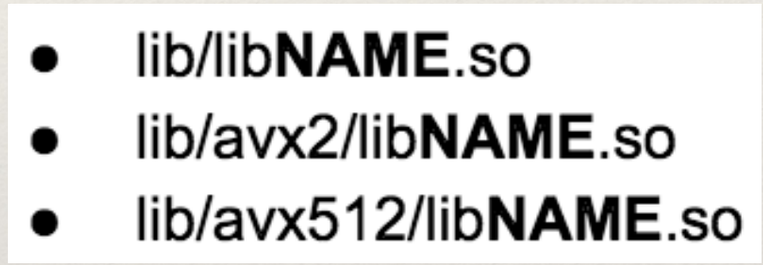
- Over **150** CMSSW Integration Builds (IBs) per week
  - **80K** files/IB
- Deployed on CVMFS **cms-ib.cern.ch**
  - Keep **two weeks** worth of IBs
  - **2TB**
  - **2** Stratum 1's
  - **12M** files/week
  - Very high publish frequency
    - **250** publish/day
    - Weekly garbage collection
- Avg. publish time < **5 minutes**
  - ARM and Power IBs takes more time
  - Mostly due to use of PRoot/QEMU





Future Plans

- Multiple instructions set CMSSW builds e.g SSE3(default), AVX2, AVX512
  - Build only selected software for multiple instructions set
  - Dynamically set runtime environment to use the best flavor
- Deployment of CI build artifacts on CVMFS
  - Allow parallel running of tests
  - Developers can use it to provide fixes
    - Specially in case of CMSSW external tools testing
  - Save a lot of build time for GPUs
- Automate creation of pre-installed CMSSW release(s) fat containers



Summary

- CVMFS and containers technologies are crucial for CMS Offline SW
  - Development, Production and SW preservation
- We are happy with CVMFS publish performance
  - Can be improved by better cataloging
    - CMSSW/src contains over 75K files
- Publishing images on unpacked.cern.ch can be improved
  - Web-hooks
  - Regular expression to install selected containers
  - Cleanup of deleted containers