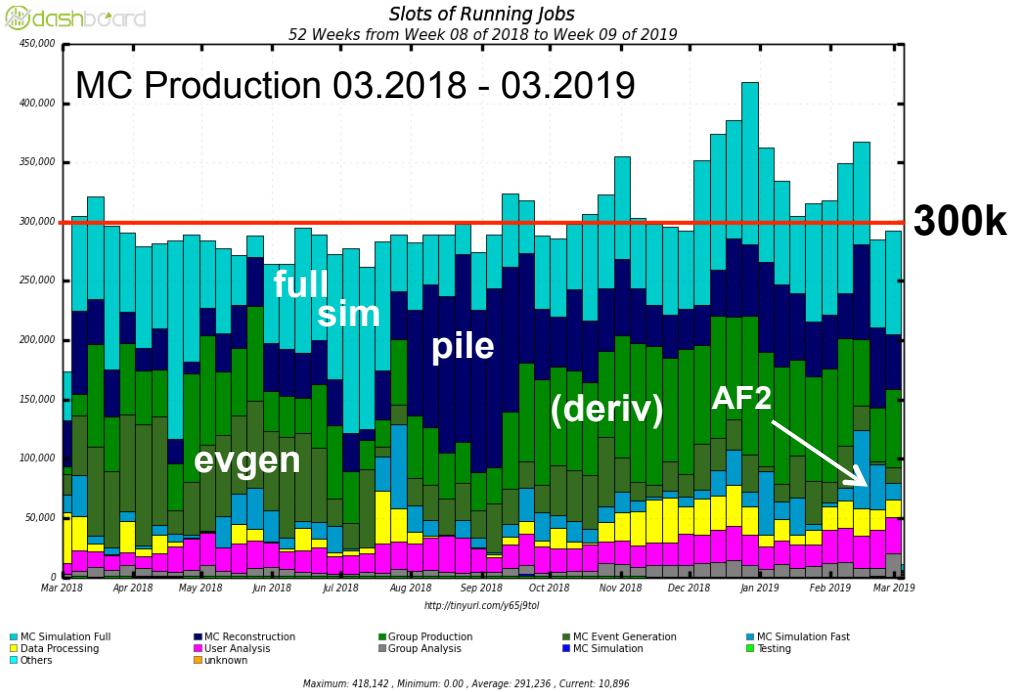
ATLAS MC Production Workflow

Input to the ATLAS Sites Jamboree

- *Introduction*
- *MC production chain*
- *MC16 workflows*
- *Single tag containers*
- *Performance in 2018*
- *Future plans and developments*



Dominic Hirschbühl (Univ. Wuppertal)
David South (DESY)

March 6th, 2019

HELMHOLTZ SPITZENFORSCHUNG FÜR GROSSE HERAUSFORDERUNGEN

ATLAS

BERGISCHE UNIVERSITÄT WUPPERTAL

DESY.

Introduction

- Mandate
  - *To produce the official MC samples in the central production system*
  - *To monitor and manage the available Grid resources for this purpose*
  - *To improve the MC production efficiency of the central production system*
- 2 production coordinators (Dominic Hirschbühl, DS)
  - *Overall coordination, interaction with other groups, as well as bulk sample production*
- 3 experienced shifters on rotation: Doug Gingrich, Hulin Wang, Matteo Negrini
  - *Produce all requests from Physics (BPHYS, EXOT, HIGG, STDM, SUSY, TOPQ) and CP Groups (EGAM, FTAG, IDET, JETM, MUON, TAUP, MCGN) with standard configurations*
- 1 upgrade production coordinator: *position vacant, but currently little activity*
- 1 special requests coordinator: Sascha Mehlhase

Interactions with other groups

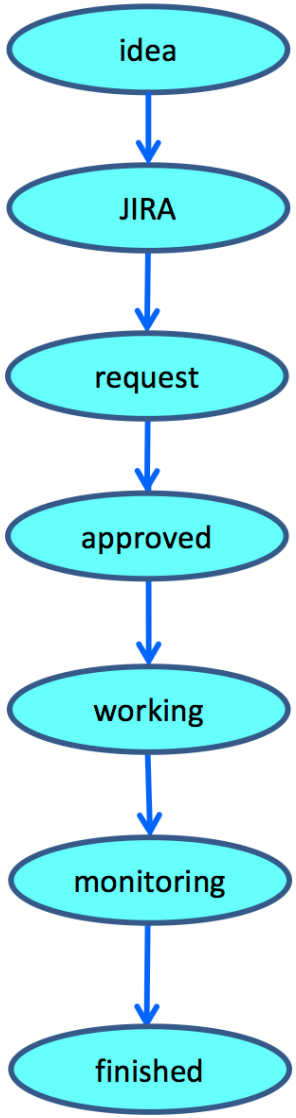
- MC is the interface between Physics and Computing

- From the Physics side:
  - *Physics Coordination (PC)*
  - *Physics Modelling Group (PMG)*
  - *Data Characterisation and Curation (DCC)*
  - *Combined Performance (CP) groups via MC Contacts*
  - *Physics Analysis (PA) groups via MC Contacts*
  - *Physics Validation*

- From the Computing and Software side:
  - *ATLAS Distributed Computing (ADC):*
    - *Workflow Management System (WFMS), ProdSys, PanDA*
    - *Metadata (AMI)*
    - *Distributed Data Management (DDM)*
    - *Non-standard resources (HPC, Cloud, BOINC, volunteer computing)*
  - *Simulation*
  - *Reconstruction*

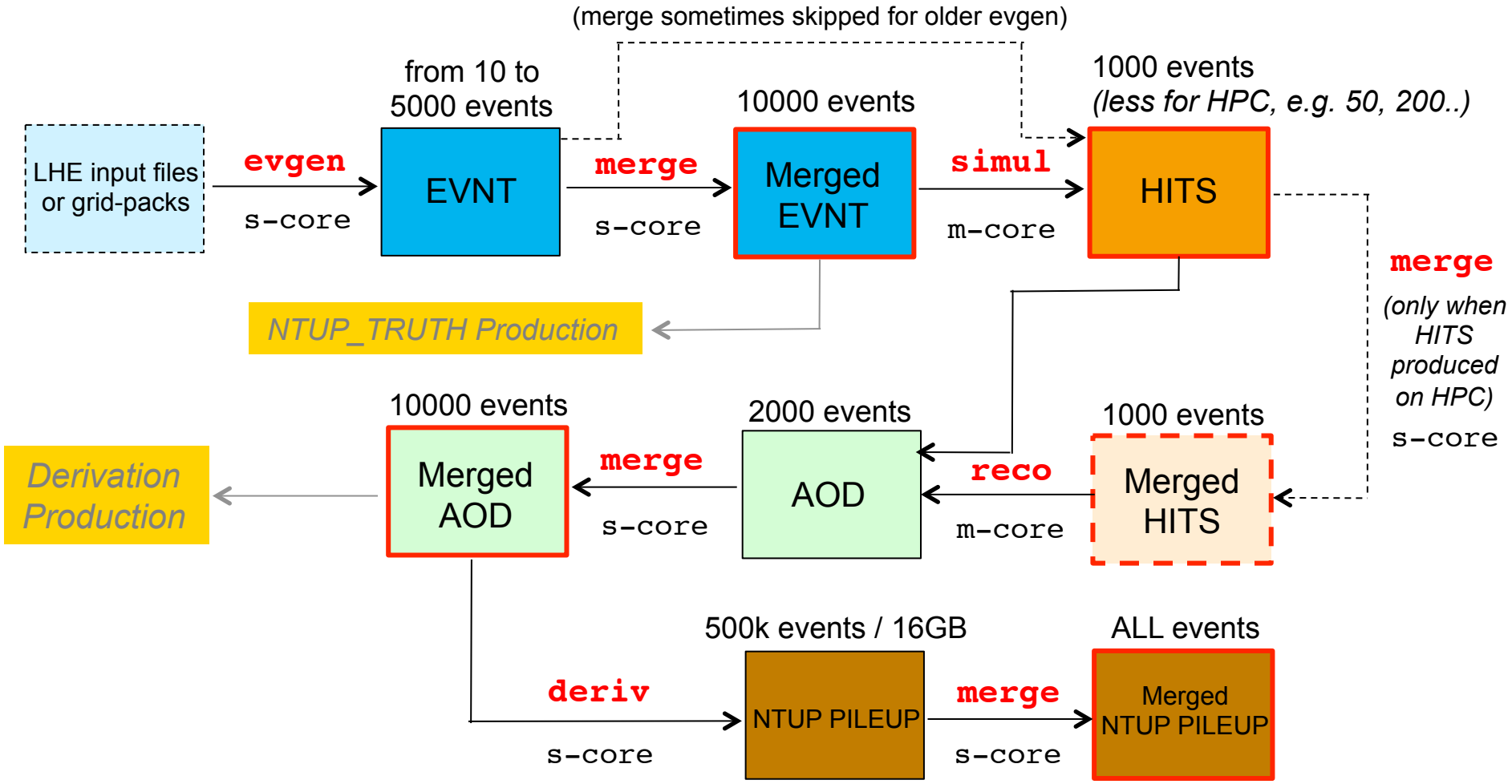
- Further contacts with trigger, data preparation, group production, …

MC production request workflow



- Requester talks to their sub-group convenor about MC needs and to their MC contact person about samples
- ATLMCPROD JIRA ticket is created, usually by MC contact
- Production request is created in ProdSys by MC contact, via input spread-sheet
- PMG convenor approves request
- MC production team member takes over the request (becomes manager), modifying it if necessary, before submitting it for production
- MC production manager monitors the request until finished
- MC production manager puts notification in JIRA when done

Current MC production chain



Keep only the (merged) dataset at each step, according to the relevant rules of the lifetime model for each data type

MC production: AMI tags

- An MC production chain consists of different production steps, each with its own tag
- All tags are maintained in AMI, main ones: `e-tag`, `s-tag`, `a-tag`, `r-tag`, `p-tag`
- `e-tags`: Evgen tag, EVNT production and merging
  - Hard scatter: *MadGraph5_aMC@NLO, Powheg, Sherpa*
    followed by Showering: *Herwig7, Pythia8, Sherpa*
  - *Sometimes with input LHE files or grid-packs. Sometimes producing LHE too. More later..*
  - *e-tag contains release, com energy and a tar.gz file of relevant (usually most recent) Job Options and as this is different for each request, many e-tags in the current (MC16) model*
- `s-tags`: Geant4 simulation to produce HITS and merging when produced on HPC
- `a-tags`: Simulation tag when running faster, less detailed AFII simulation
- `r-tags`: Digitisation and reconstruction, as well as AOD merging
- `p-tags`: Production of NTUP_PILEUP format and merging
  - *Used by analysis in conjunction with merged AOD, contains same events*

The MC16 campaign

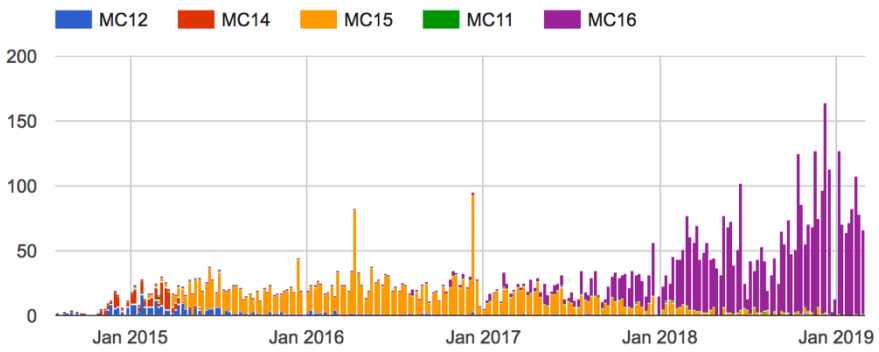
Sub-campaign	Data match	Processing	Description
MC16a	2015+2016	HITS+digi+reco	matches 2015+2016 dataset + mu profile
MC16b	none	digi+reco	uses higher mu (30-70) for trigger and CP studies for 2017 data
MC16c	2017	HITS+digi+reco	uses expected 2017 mu profile, geometry and trigger
MC16d	2017	digi+reco	uses true 2017 mu profile
MC16e	2018	HITS+digi+reco	uses expected 2018 mu profile and trigger

- There were six **sub-campaigns** foreseen in MC16, MC16a-f, in reality only 4 are used
- MC16a/c/e comprise of statistically different EVGEN events in the same configuration, so they can be combined for analyses using all Run 2 data
- The simulation configuration for HITS is also the same for all MC16a/s/e sub-campaigns
- MC16a/c/e are initial reco versions for 2015/16, 2017 and 2018 with the initial mu profile
  - *MC16c superseded by MC16d, with updated pile-up distribution for 2017*
  - *Pile up profile estimate accurate enough in 2015/16 and 2018 meaning no new reco needed*
  - *New reco sub-campaigns like MC16d may also employ different conditions, e.g. if part of some detector got disabled during the run and is thus masked in the new MC version*

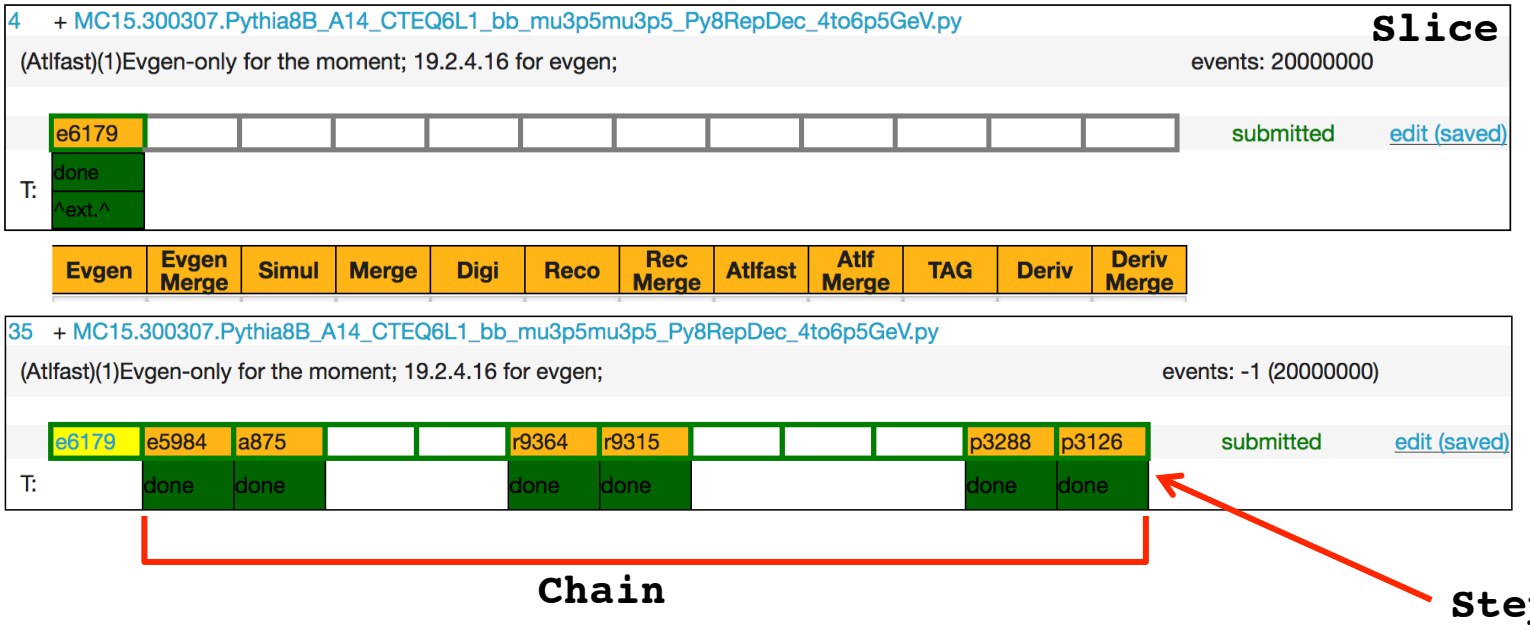
What does this mean for MC16 production?

Physics	evgen		simul		recon	
Sub-campaign	Project	Campaign	Project	Campaign	Project	Campaign
MC16a	mc15_13TeV	MC16:MC16a or MC15:MC15* or None	mc16_13TeV	MC16:MC16a	mc16_13TeV	MC16:MC16a
MC16b	mc15_13TeV	MC16:MC16b	mc16_13TeV	MC16:MC16b	mc16_13TeV	MC16:MC16b
MC16c	mc15_13TeV	MC16:MC16c	mc16_13TeV	MC16:MC16c	mc16_13TeV	MC16:MC16c
MC16d	mc15_13TeV	MC16:MC16c	mc16_13TeV	MC16:MC16c	mc16_13TeV	MC16:MC16d
MC16e	mc15_13TeV	MC16:MC16e	mc16_13TeV	MC16:MC16e	mc16_13TeV	MC16:MC16e

- Multiple configurations to cover the different Run 2 running periods covered by MC16
- The evgen step has project mc15_13TeV, originally because this uses to an earlier release
- All others, **including evgen merge**, have project mc16_13TeV
- Also note that for MC16a we are sometimes using older evgen produced in the MC15 campaign
- In the case of MC16d, only the reco (with the updated pile-up) has that sub-campaign, the HITS are still MC16c
- This complexity, as well as the higher integrated luminosity, contributes to MC16 being significantly larger than previous campaigns



MC16a workflow: 2 separate requests



Request 1

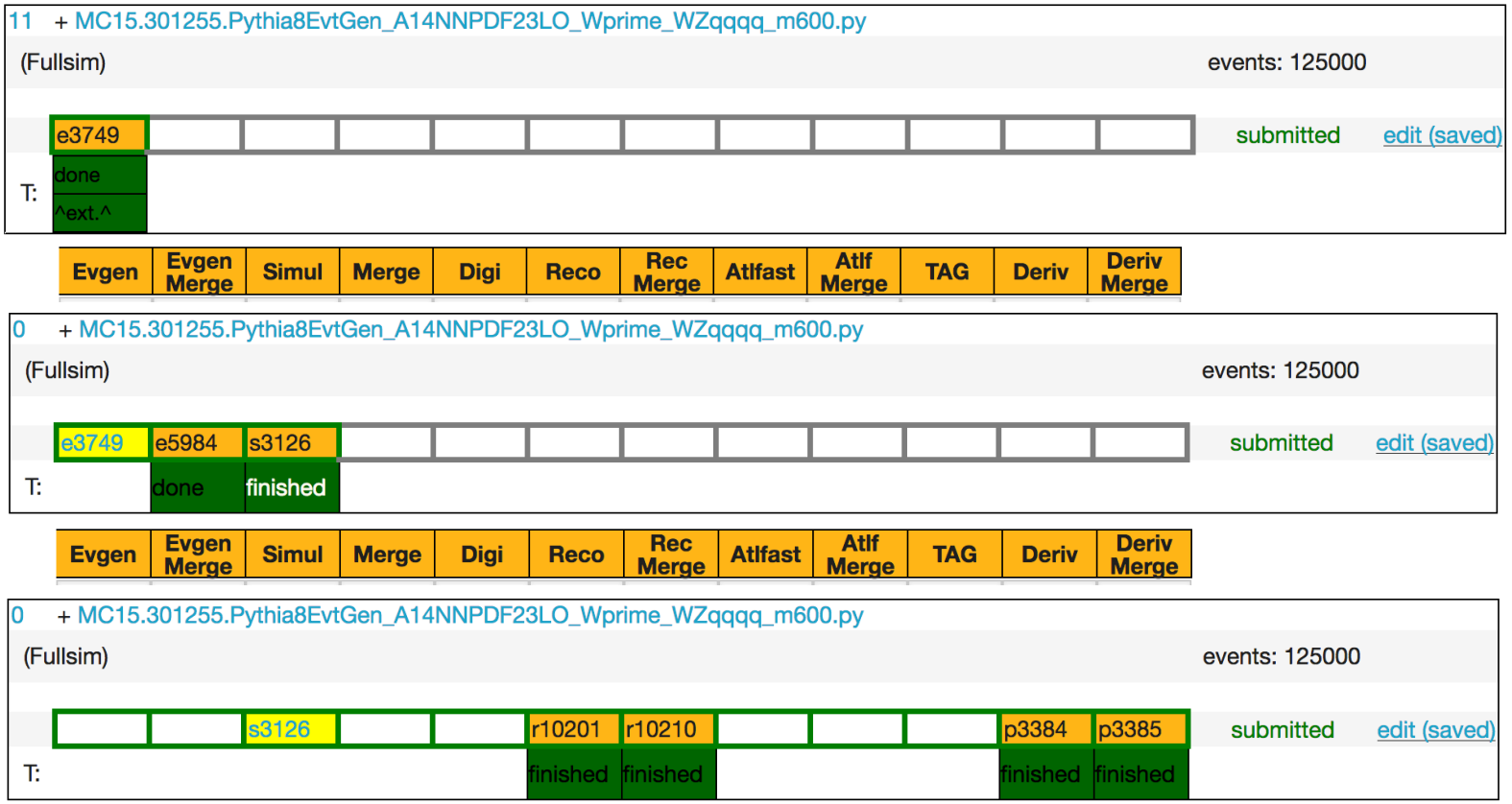
Project:
mc15_13TeV
Sub-campaign:
MC15c/MC16a/None

Request 2

Project:
mc16_13TeV
Sub-campaign:
MC16a

- Project is constant within a request, so here across one slice
- For both MC16a and MC16c, the standard workflow is 2 requests:
  - *First request performs the Evgen step*
  - *Second request to do the Evgen merge, (Fast)Simul, Reco+Merge and Deriv+Merge steps*
- This example is for MC16a: the workflow is similar for MC16c (and MC16e), which uses the MC16c (MC16e) sub-campaign for both request 1 and 2

MC16d workflow: 3 separate requests



Request 1

Project:
mc15_13TeV
Sub-campaign:
MC16c

Request 2

Project:
mc16_13TeV
Sub-campaign:
MC16c

Request 3

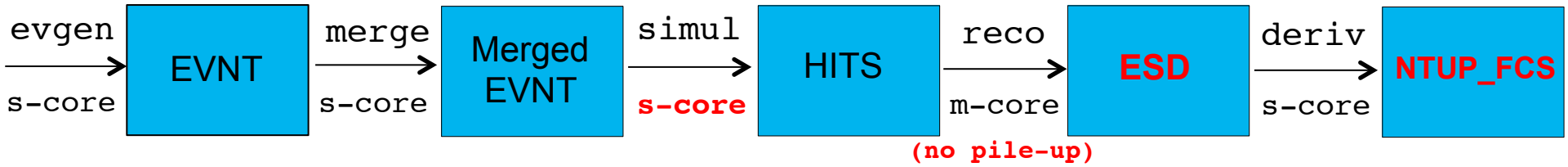
Project:
mc16_13TeV
Sub-campaign:
MC16d

- The standard MC16d workflow for new requests necessarily involves three requests
  - *The first request contains the evgen step*
  - *The second request contains the evgen merge and simulation steps*
  - *The third request contains the reco, reco merge, deriv and deriv merge steps*

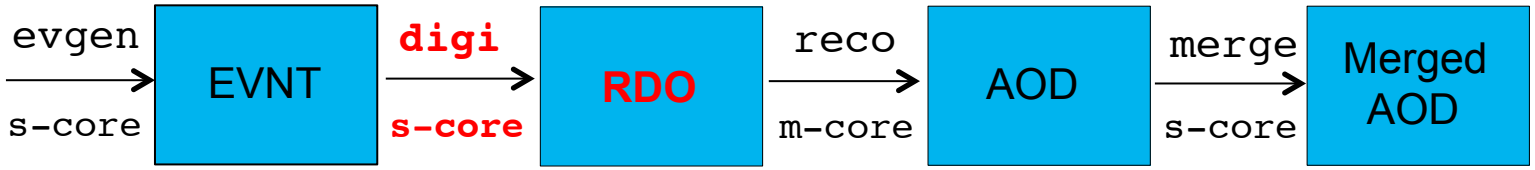
Alternative workflows

- On top of the bulk campaigns, there are always a large number of small requests, some of which are standard, some highly specialised

- A recent example is the on going *FastCaloSim* production, producing many thousands of slices of single particles in a special format



- Another is the HI request with full simulation production of dijet samples over-layed with the minimum bias 5.44 TeV XeXe data events



Single-tag containers

- The development of the single tag container model for MC16 was almost inevitable for MC Production, due to:
  - *The variation in the number of merge tags in evgen, from using older HITS productions created before evgen was universally merged – thankfully now much rarer*
  - *The variation in the number of merge tags in simulation, where for HPC the lower number of events/job requires an additional merge step*
  - *Extending samples, where original and extension may involve alternative workflows*
  - *And to some extent, now having tids from multiple sub-campaigns within one container*
- The single-tag concept is applied to evgen, simul, recon and deriv containers
- First concerning evgen and simulation:

`evgen.EVNT.e/`
containers have (unique) tids of the format:

`evgen.EVNT.e`
`merge.EVNT.e_e`

`simul.HITS.e_s/`
containers have (unique) tids of the format:

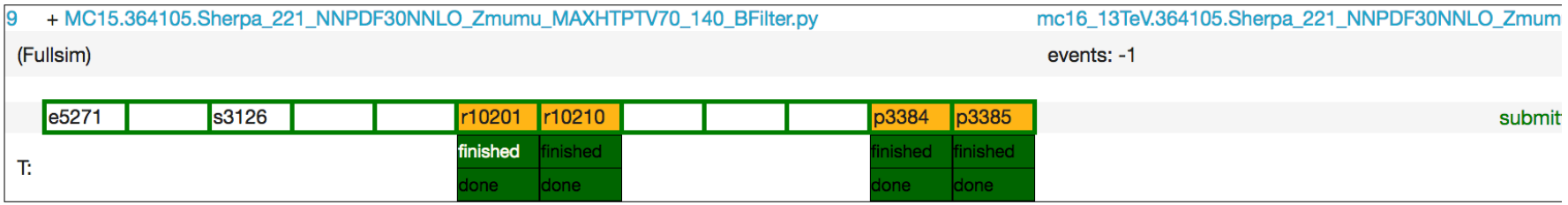
`simul.HITS.e_s`
`merge.HITS.e_s_s`
`simul.HITS.e_e_s`
`merge.HITS.e_e_s_s`

An example of a varied MC16 HITS single tag container

mc16_13TeV.364105.Sherpa_221_NNPDF30NNLO_Zmumu_MAXHTPTV70_140_BFilter.simul.HITS.e5271_s3126/

Dataset Name	Events Number	SubCampaing	Tasks
mc16_13TeV:mc16_13TeV.364105.Sherpa_221_NNPDF30NNLO_Zmumu_MAXHTPTV70_140_BFilter.simul.HITS.e5271_e5984_s3126_tid12196360_00	1489400	MC16:MC16c	13038485
mc16_13TeV:mc16_13TeV.364105.Sherpa_221_NNPDF30NNLO_Zmumu_MAXHTPTV70_140_BFilter.simul.HITS.e5271_e5984_s3126_tid12592858_00	7431200	MC16:MC16e	
mc16_13TeV:mc16_13TeV.364105.Sherpa_221_NNPDF30NNLO_Zmumu_MAXHTPTV70_140_BFilter.simul.HITS.e5271_e5984_s3126_tid13866273_00	2510000	MC16:MC16e	
mc16_13TeV:mc16_13TeV.364105.Sherpa_221_NNPDF30NNLO_Zmumu_MAXHTPTV70_140_BFilter.simul.HITS.e5271_s3126_tid10730514_00	1995000	MC16:MC16a	
mc16_13TeV:mc16_13TeV.364105.Sherpa_221_NNPDF30NNLO_Zmumu_MAXHTPTV70_140_BFilter.simul.HITS.e5271_s3126_tid10944971_00	3986600	MC16:MC16a	
mc16_13TeV:mc16_13TeV.364105.Sherpa_221_NNPDF30NNLO_Zmumu_MAXHTPTV70_140_BFilter.simul.HITS.e5271_s3126_tid11324488_00	5981200	MC16:MC16c	13038493

- MC16 EVNT and HITS are identical configuration, so in this container there are:
  - Two MC16a tids, *where the evgen was not merged, so only one e-tag*
  - Two MC16c tids (highlighted), *one with an evgen merge (two e-tags) and one without*
  - Two MC16e tids, *both with evgen merge steps*
- This is **extremely useful** for production and within an MC16d reco request the two MC16c HITS tids are identified and processed together:



Single tag containers: Reco and beyond

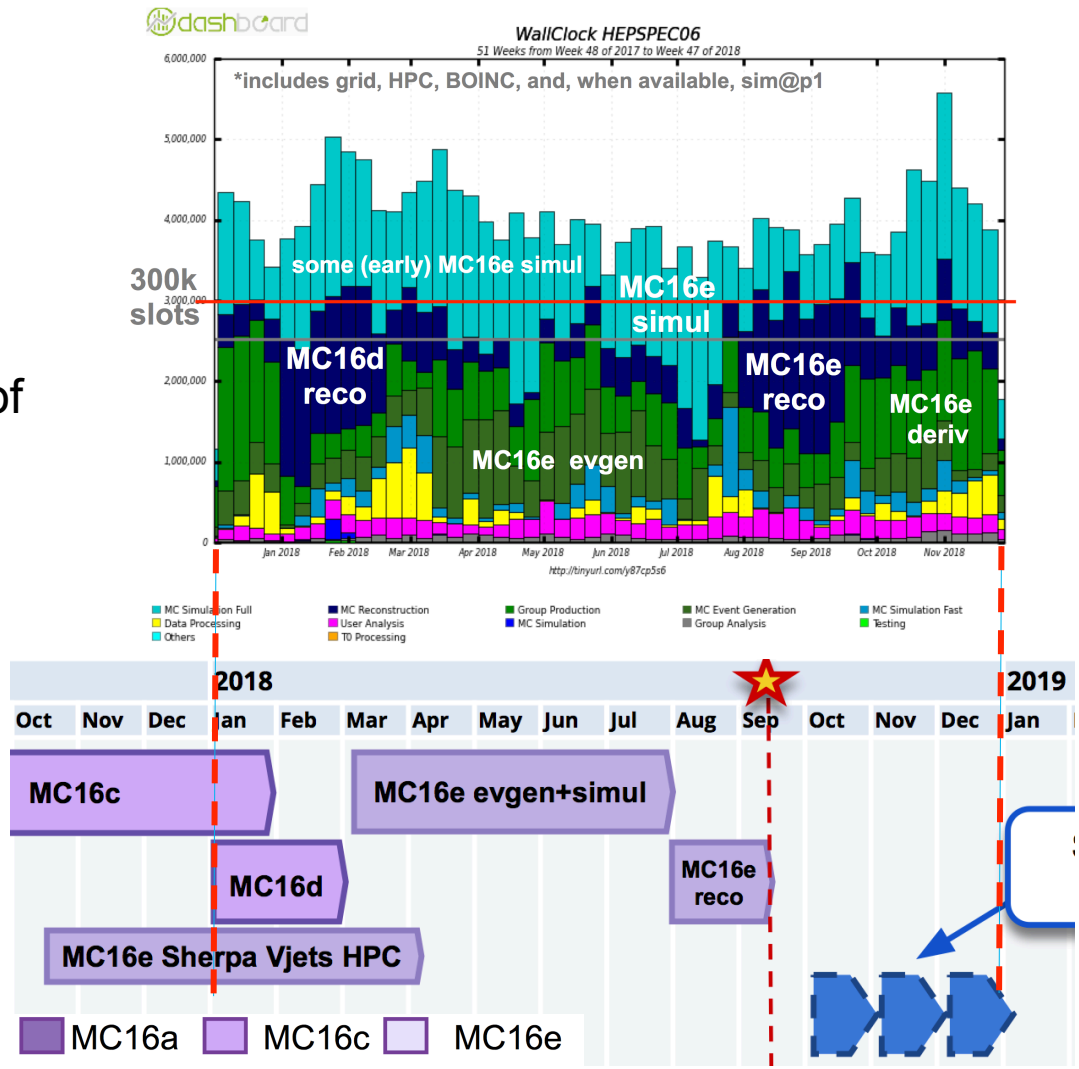
- For merged AOD containers in MC16, the following patterns are therefore relevant (tags detailed on twiki, e.g https://twiki.cern.ch/twiki/bin/viewauth/AtlasProtected/AtlasProductionGroupMC16e)
  - MC16a: **`recon.AOD.e*_s3126_r9364/`**
    MC16d: **`recon.AOD.e*_s3126_r10201/`**
    MC16e: **`recon.AOD.e*_s3126_r10724/`** (and for AF2, replace **`s3126`** with **`a875`**)

"**`recon.AOD.e_s_r`**"
Production step: recon
One tag of each type

- Some additional points on single-tag containers:
  - *The content of **`recon.AOD`** and **`deriv.NTUP_PILEUP`** containers is simpler, as we always merge these formats: they only ever have "merge" tids after all merges are done*
  - *Derivations use internal merging, so all tids in other **`deriv.NTUPXXX`** containers are "deriv"*
  - *Data produces many formats from RAW, so in that case it's not just **`recon.AOD`** but also **`recon.DRAW_RPVLL`** etc, so both production step and format are important*

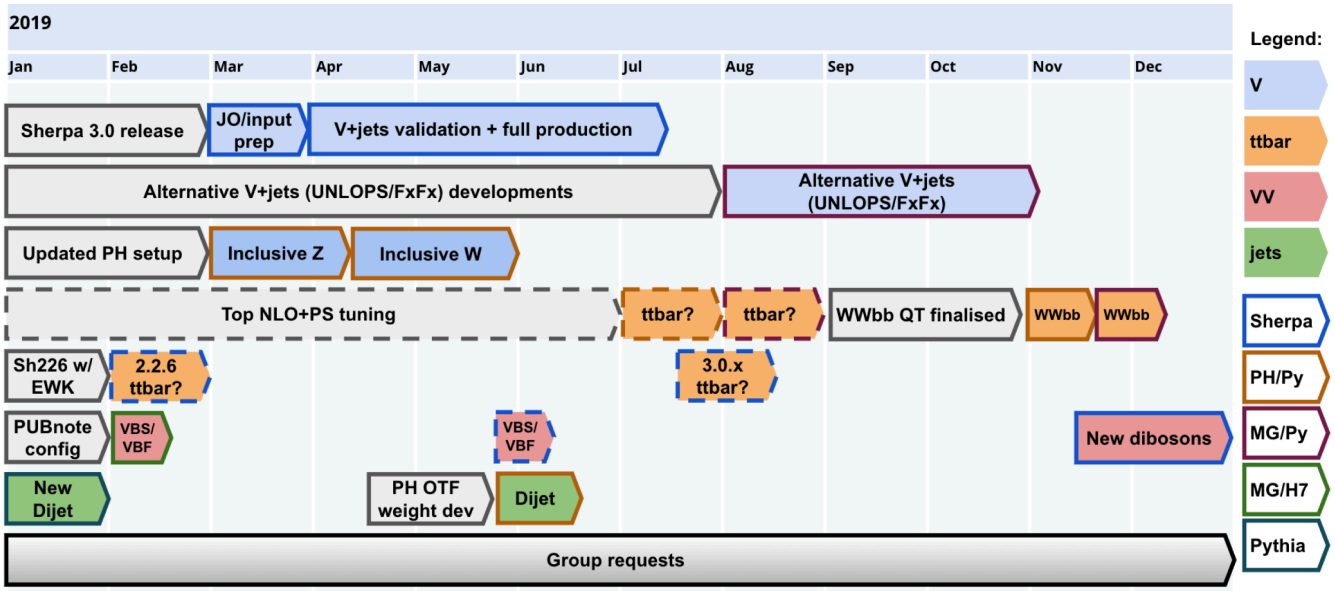
2018: MC16 MC production summary

- Successful MC16e production throughout 2018, resulting in AODs being available for derivations on if not ahead of schedule
- MC16e pile up profile within 5% of the actual distribution in data, no need this year for large scale re-reco of the MC16e HITS as an MC16f production
- Current stats for MC16 events:
  - *MC16a: FS=4.93B AF=1.42B*
  - *MC16d: FS=5.37B AF=1.16B*
  - *MC16e: FS=4.03B AF=1.13B*

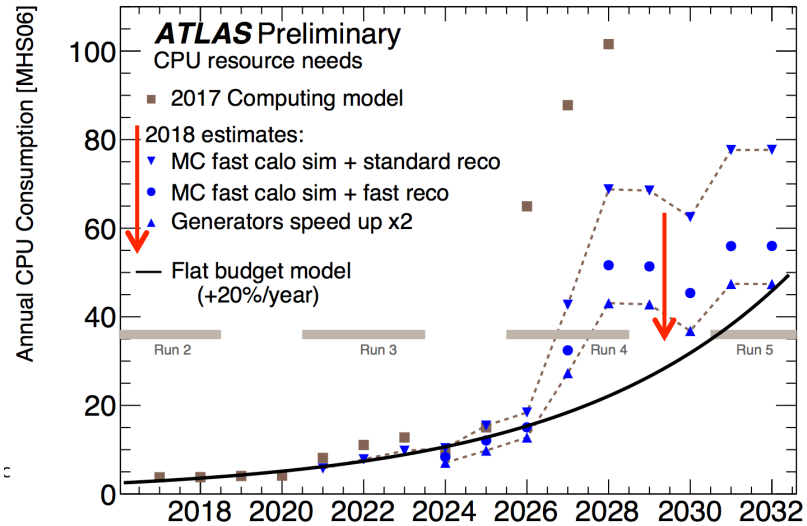


Future plans: Generators and EVNT

Josh McFayden

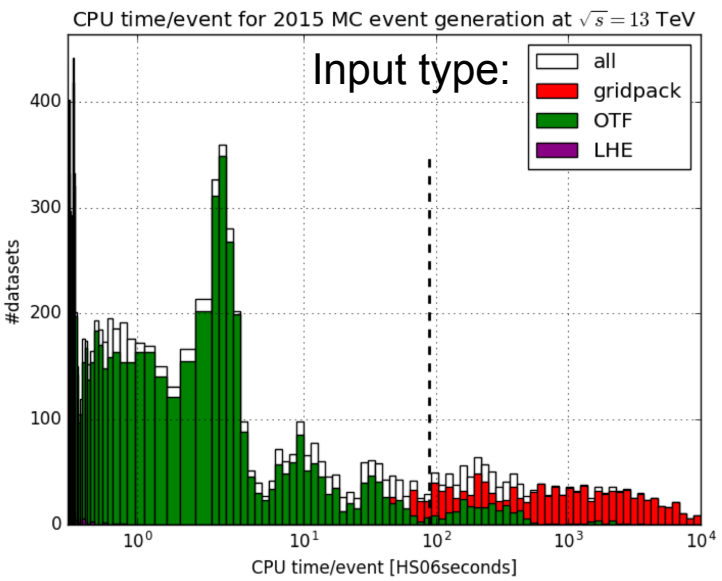


- We're now in a shutdown period
- 2019 production plans are different, based on new/updated generators
- Much effort on going to reduce CPU requirements given that current model does not scale beyond run 3
  (details: Josh's talk from Feb ATLAS Week)



Future plans: LHE events

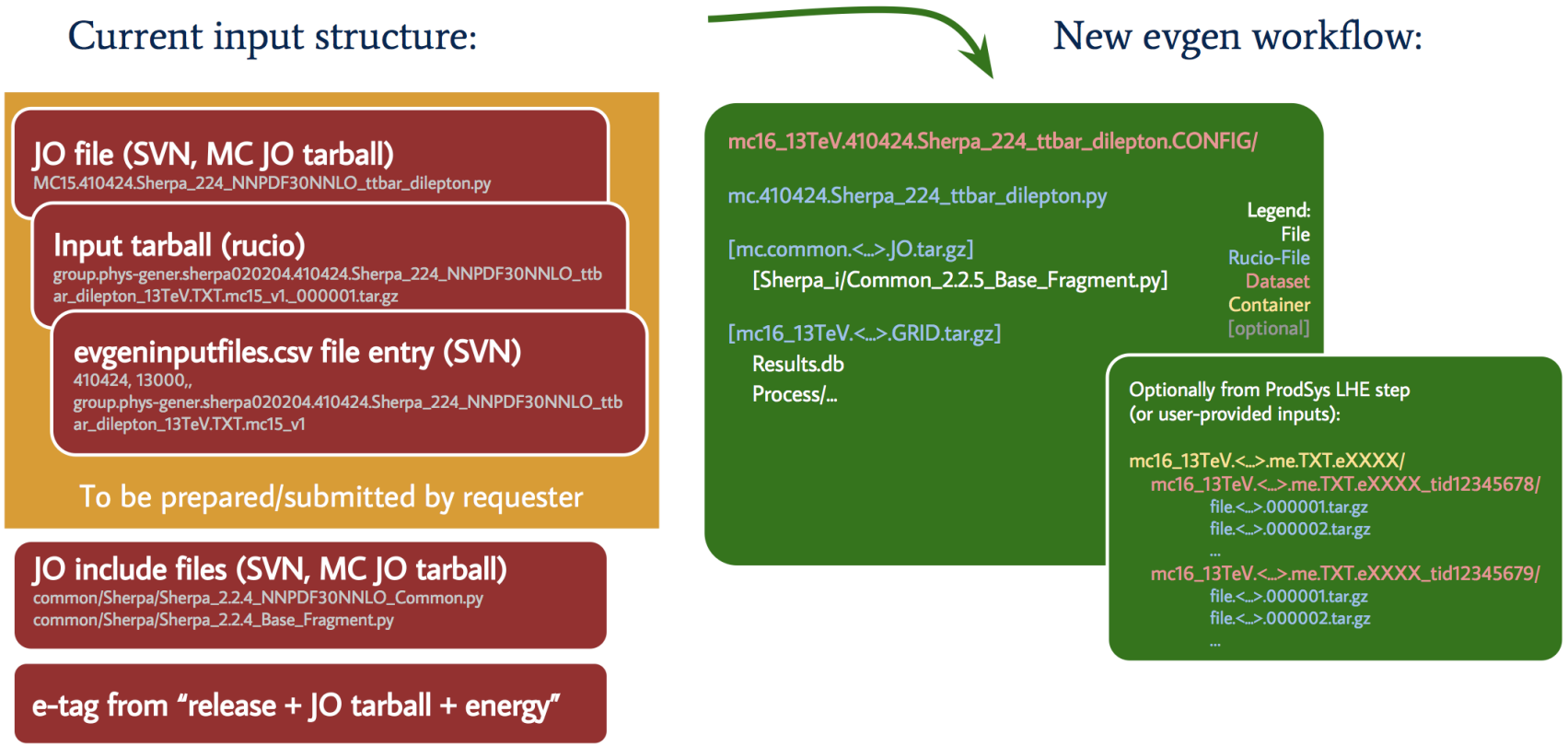
- The use of Les Houches Event (LHE) files, datatype "TXT", as input to EVNT jobs is becoming more common and will likely be expanded in the future
- Saving this allows the same hard scatter events to be re-showered with a different model, e.g. for a systematic study, also saving CPU



- In addition, for ttbar we also have three very large samples, with tasks of 48k jobs, each producing a file of 16.5k event used for filtered samples
  - This workflow is likely to be expanded into the Higgs samples
- This means rucio containers with > 3M files, way too large to loop over, and results in ugly, many sub-slice evgen requests you may have seen
  - In the future, we plan to use temporary containers to group the input tids together to avoid such a large number of slices

Future plans: Remodelling evgen

Frank Siegert



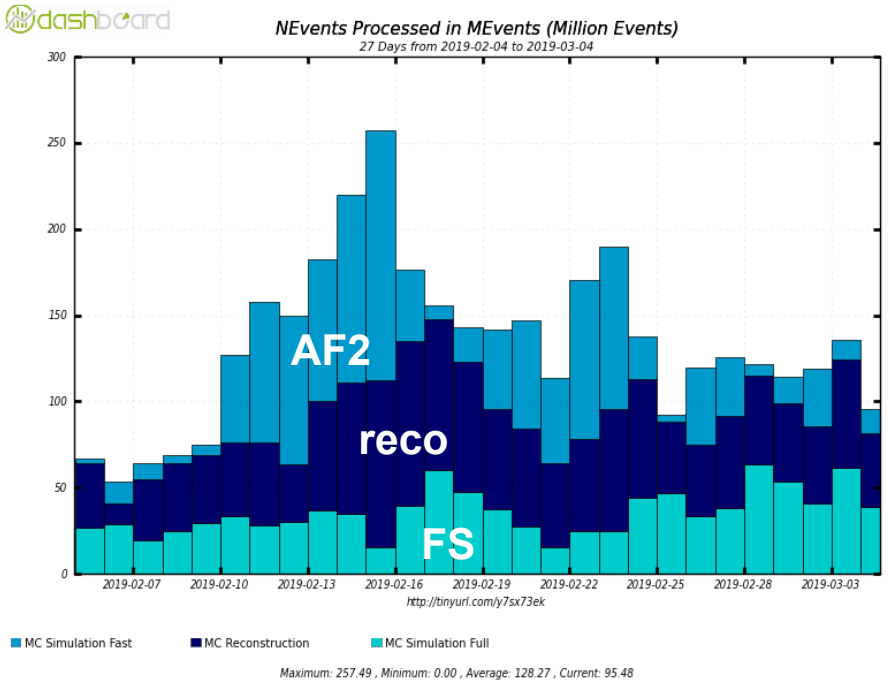
> For the next campaigns anticipate that the evgen workflow is simplified

- *Removal of SVN dependency, large reduction in number of e-tags, fewer tar-balls*
- *Now rely on rucio container to define configuration and JOs, e-tag only release based*
- *Cleaner LHE production step, without EVNT, via* **`Generate_tf`** *adaptation*

Future plans: Fast(er) simulation

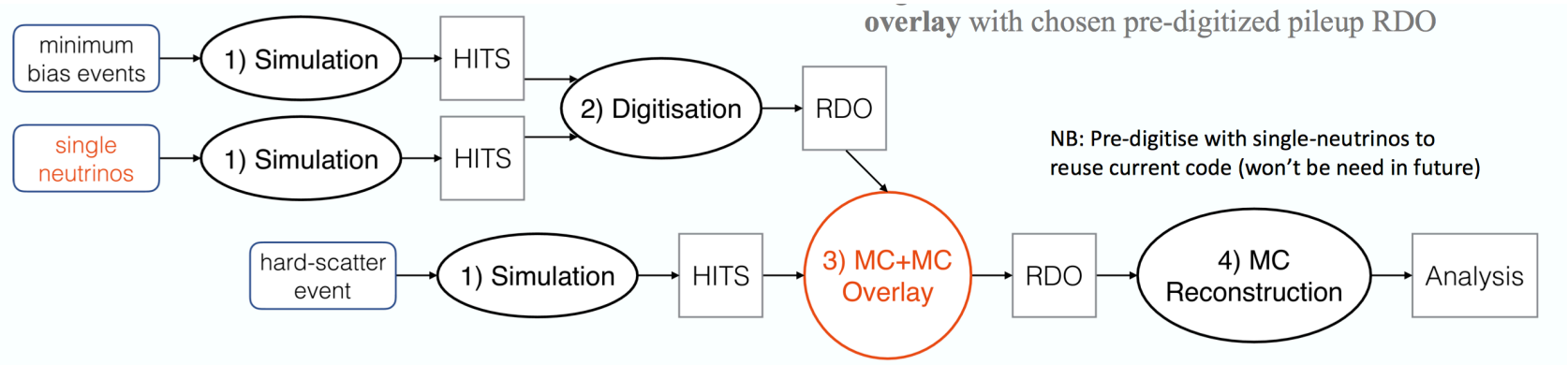
- Simulation remains the main CPU user, daily output of around 30M events
- Fast simulation allows many more events to be produced for less CPU - at the price of detail in the simulation
- Will be necessary to do more fast simulation in the future, FastSimV2 in physics validation now
- Efforts also on going in to run full simulation with AthenaMT
- Details: Hasib's talk from Feb ATLAS Week

	Avg CPU/evt
Evgen	~80 s
FullSim	~245 s
FastSim	~45 s
Reco	~60 s

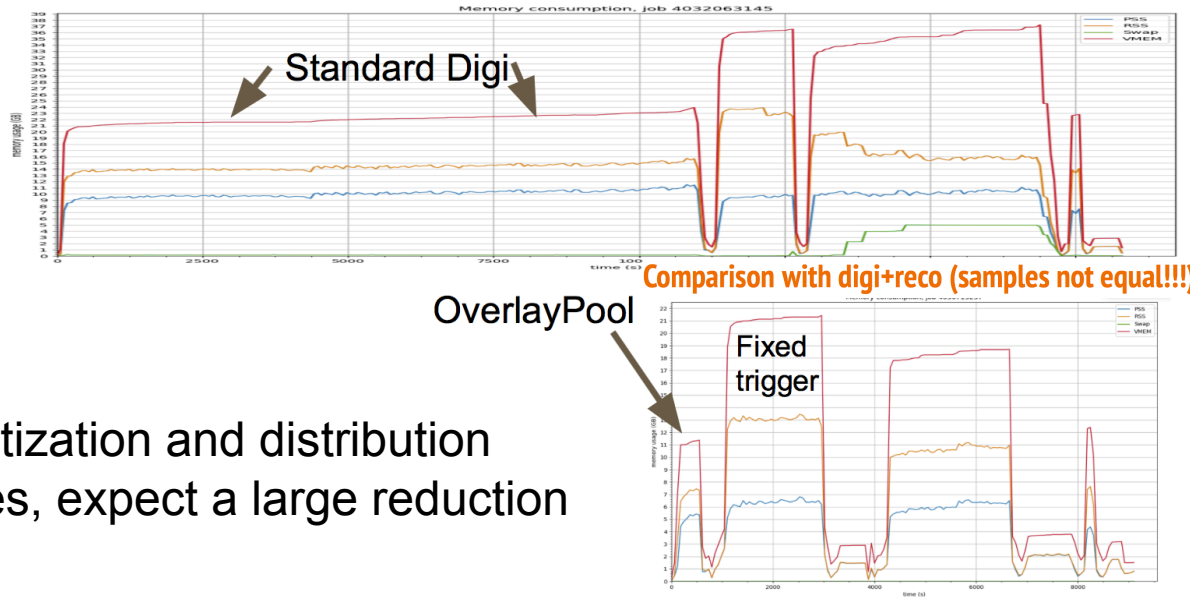


Future plans: New pile-up overlay workflow

Andrej Filipcic

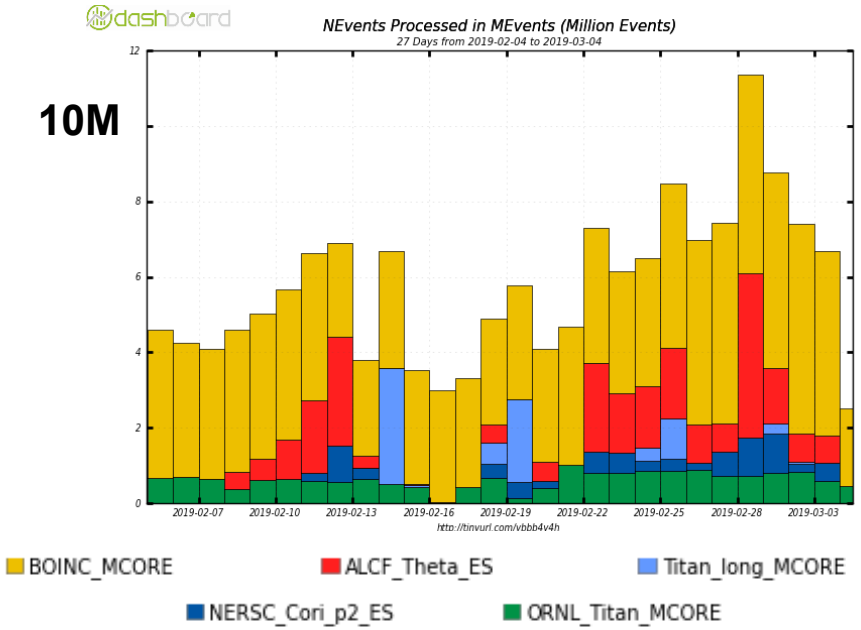


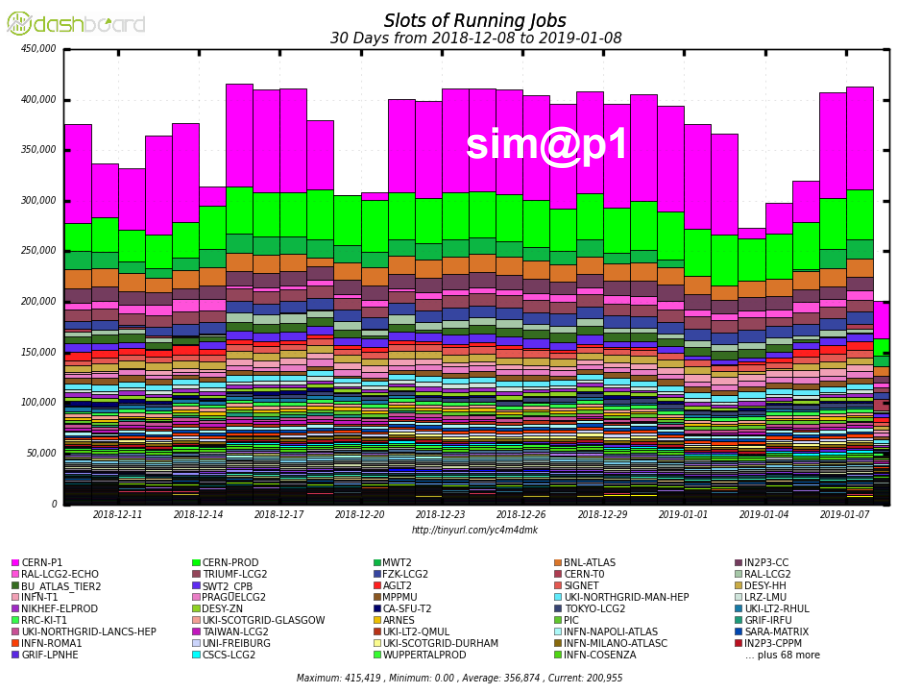
- To use pre-digitized MB events instead of digitizing the hard scatter + MB every time
- From latest talk I find (OAB 24.02): Overlay software is now fully validated, pending final verification of metadata, muon truth matching & pixel hole issue



- End result: After creation, digitization and distribution of large minimum bias samples, expect a large reduction in CPU time for reco jobs

HPC, opportunistic resources





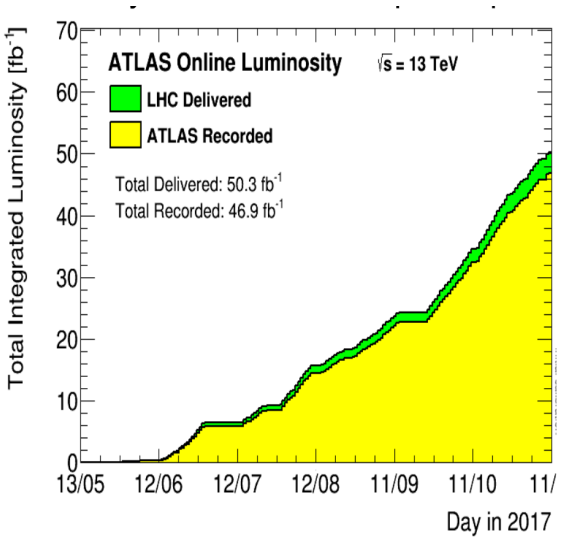
- I don't really dare mention HPC, given the last two days, but here are some plots
- BOINC + Titan backfill are constant, bursts from other sites: significant fraction of FS
- Huge amount of simulation done on the HLT farm over Christmas
- Future workflow: continue! Jumbo and Co-Jumbo Jobs and Event Service jobs: see Tadashi's talk from this morning

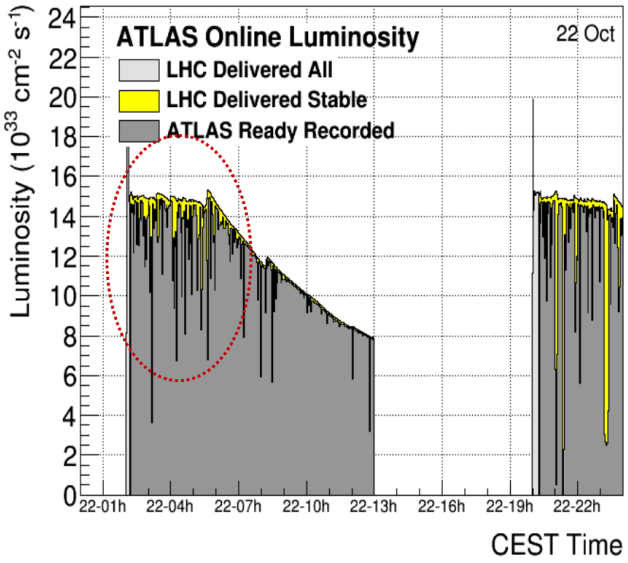
Summary

- The MC16 workflow has proven to be robust, despite of / thanks to a lot of necessary innovation in ProdSys and beyond, applied as we go along
  - *The novel, non-trivial handling of multiple sub-campaigns, multiple tids within those sub-campaigns and varying numbers of single and double tags has also shown to be successful*
- Some areas are going to change and present challenges in the coming years
  - *Evgen: Fewer e-tags, new rucio-based config, more use of LHE inputs, m-core (I/O?)?*
  - *Simulation: More FastSim(V2) expected; more Event Service jobs*
  - *Reconstruction: New pileup pre-mix expected to reduce CPU usage*
  - *(Co-)Jumbo Jobs: unifying jobs within a task across heterogeneous resources*
- There are areas where we can optimise the current production model
  - *Consider internal merging like is done in the derivations?*
  - *Running later steps in the chain in a more coordinated way, shorten the tails*
  - *Multiple projects and sub campaigns in one request? Linking requests together?*
  - *Discussions still on going within DCC about future nomenclature, improvements for user*

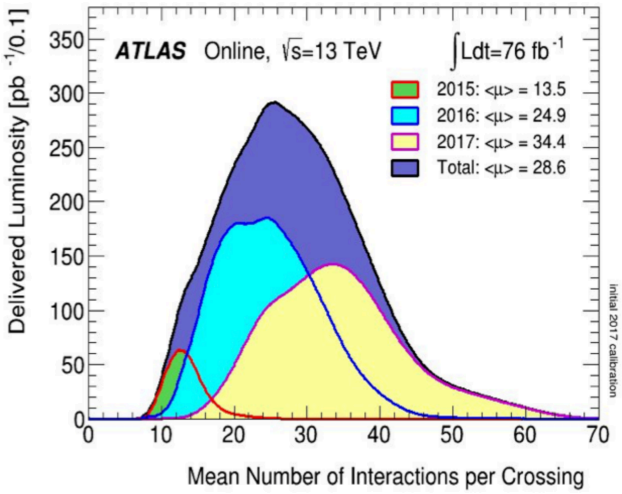
Backup

Pile-up in 2017 data taking: the need for MC16d

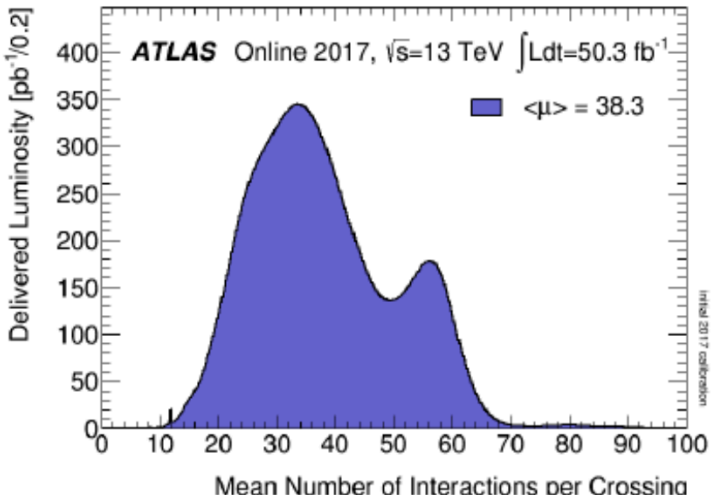




Initial 2017 pile-up profile (yellow)



(final) 2017 pile-up profile



- Successful 2017 data taking in terms of recorded luminosity, but challenging later in the year with novel 8b4e bunch structure
- Luminosity leveling at the beginning of each run meant significant and steady amount data with pile-up just below 60
- Clear need for MC16d to match the real data taking conditions

Nomenclature, production steps and formats

- MC Production is wide reaching in the ATLAS computing infrastructure, influencing and fully integrated with ProdSys, rucio, AMI and beyond

- ATLAS dataset nomenclature is defined in (an evolving) document https://cds.cern.ch/record/1070318, section 6 which defines MC dataset names
  project.datasetNumber.physicsShort.prodStep.dataType.AMITag[_tidnnnnnn[_SS]

- In particular section 6.1.3 describes at length the well defined options and names for the **prodStep** and the associations with e.g. the ami tags used

- The production steps used in MC Prod are: **evgen**, **simul**, **recon**, **deriv** where deriv in our case refers only to the NTUP_PILEUP format

- **merge** steps are also described, decision was taken in ATLAS that dataset containers used for analysis, or as input to analysis, should no longer contain "merge" but should have the production step associated to that format
  – e.g. "recon" for AOD

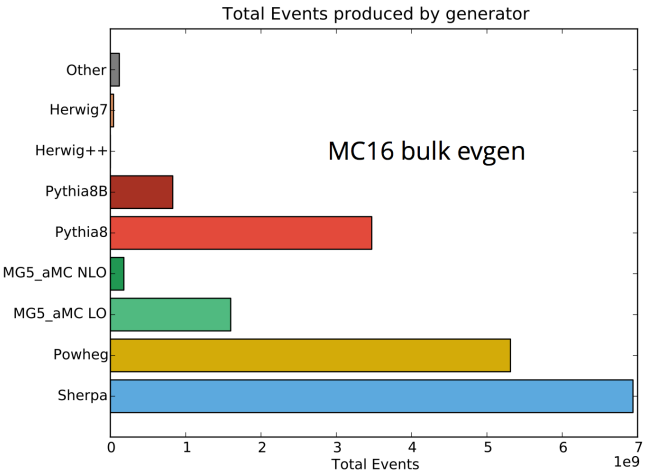
The input container for derivation productions

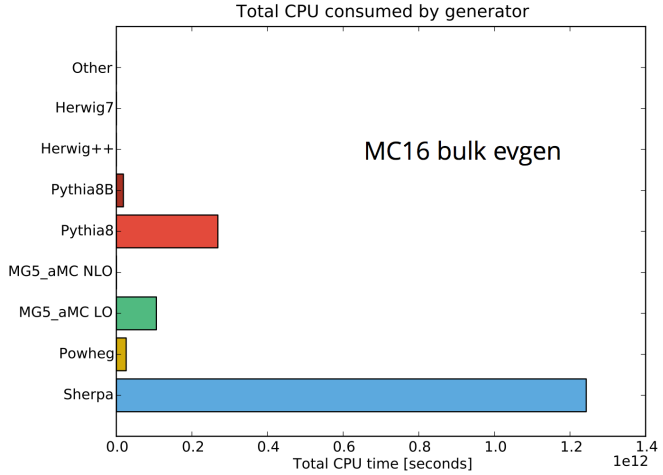
1) People should not have to care about tids, only containers
2) The tids in one container are unique, i.e. there is no double counting
3) The idea is to have **one** container for all types of tids in one sample, with and without e/s merges
4) Merges are really not very interesting, so the only tags of significance are evgen, simul, recon
5) Therefore the production step for the container should be evgen, simul, recon and there is one container per sample (i.e. per DSID)
6) The rule for the derivation input container nomenclature is simple:

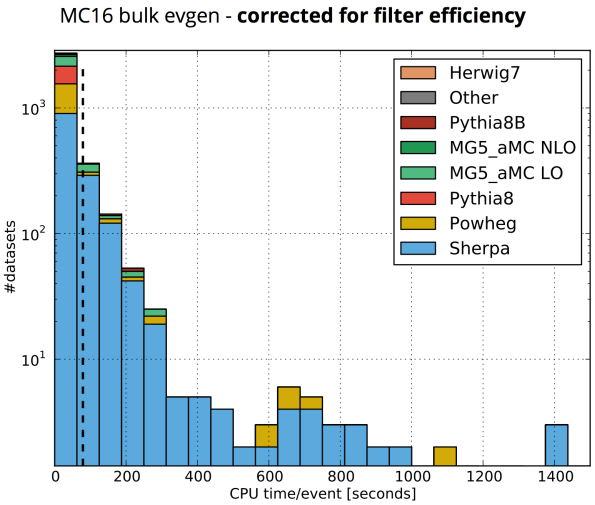
"`recon.AOD.e_s_r`"
Production step: recon
One tag of each type

Future plans: Generators and EVNT

Josh McFayden







- Number of events from Powheg and Sherpa are fairly similar
- Majority of CPU consumption comes from Sherpa2.2 V+jets setups
  - By far the largest (3.2B events) samples & most precise (V+0,1,2j@NLO+3,4j@LO)
- Average across all samples is 80s/event, but there are long tails
  - Some examples where event generation is slower than full simulation: room for improvement
- Further details in Josh's talk from Feb ATLAS Week

Long list of tids possible

146 + MC15.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.py — mc16_13TeV.364156.Sherpa_221

(Fullsim) — events: -1

	e5340		s3126			r10201	r10210				p3384	p3385
T:						done	done				done	done
						done	done				failed	aborted
						done	done				done	done
						done	done				done	done
						done	done				done	done
						done	done				done	done
						done	done				done	done
						done	done				done	done
						done	done				done	done

Long list of tids possible

146 + MC15.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.py mc16_13TeV.364156.Sherpa_221

(Fullsim) events: -1

Dataset Name	Events Number	SubCampaing
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483117_00	1999900	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483123_00	1996300	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483128_00	1995800	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483136_00	1995900	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483142_00	1999850	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483149_00	1999950	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483154_00	1999800	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483161_00	1999950	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483166_00	1999950	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483172_00	1999700	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483177_00	1999800	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483184_00	1999950	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.merge.HITS.e5340_e5984_s3126_s3136_tid11483204_00	990000	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.simul.HITS.e5340_e5984_s3126_tid12197119_00	6217000	MC16:MC16c
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.simul.HITS.e5340_e5984_s3126_tid12944773_00	31098000	MC16:MC16e
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.simul.HITS.e5340_s3126_tid10730390_00	8330000	MC16:MC16a
mc16_13TeV:mc16_13TeV.364156.Sherpa_221_NNPDF30NNLO_Wmunu_MAXHTPTV0_70_CVetoBVeto.simul.HITS.e5340_s3126_tid10944745_00	16469000	MC16:MC16a