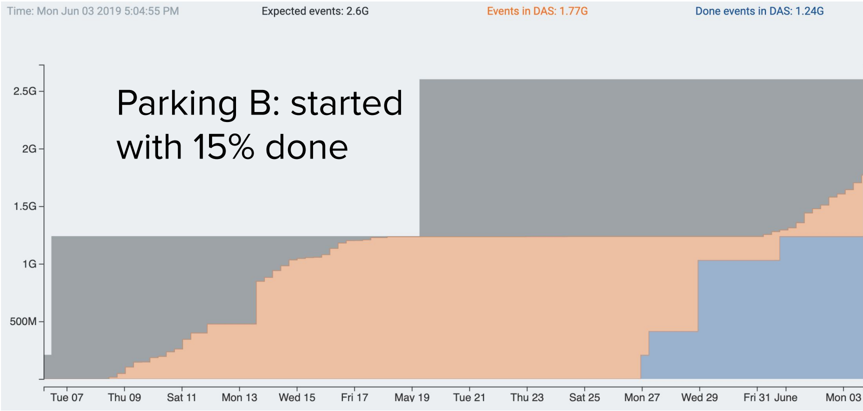
CMS Status Report

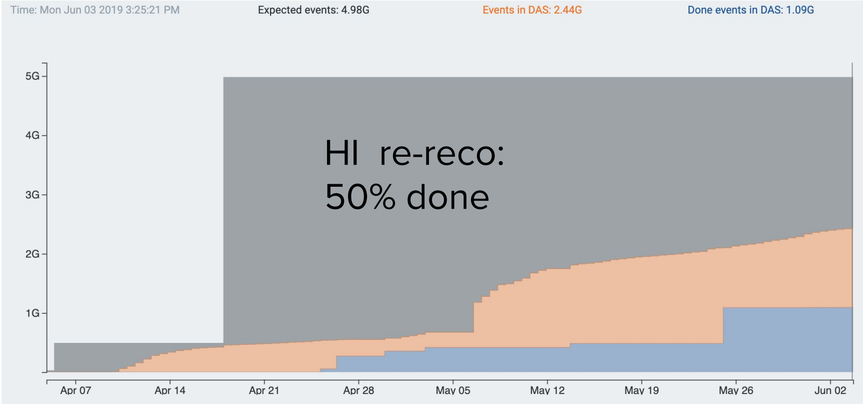
LHCC WLCG Meeting - June 4th, 2019

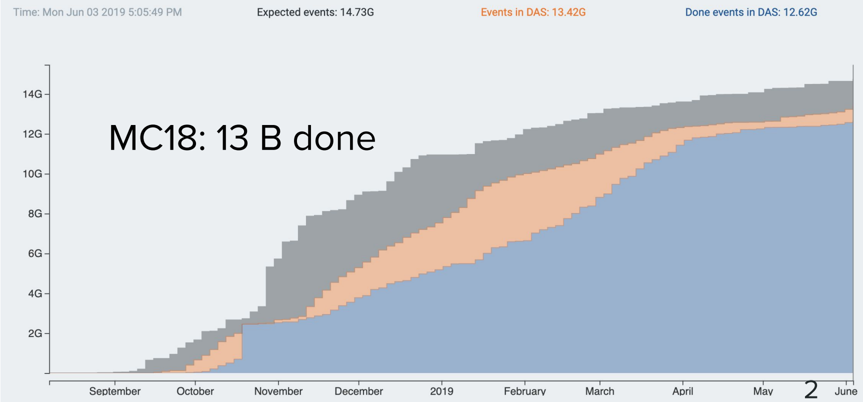
Tommaso Boccali (INFN/CERN), Markus Klute (MIT)

Status of operations

- **The first part of the year was planned to mostly host**
  - Parking B samples: once low electron Pt code ready, run on the 12B events
  - HI samples: a complete rerun needed in order to improve ECAL calibration
  - Provide MC for Moriond19 and LHCP2019 analyses
  - Set up Ultra Legacy processing, and start with 2017 data year
  - Start preparing samples for the L1 TDR
  - Complete a program of Global Runs to test new features @ P5
- All started. Many on time, some with typical delays

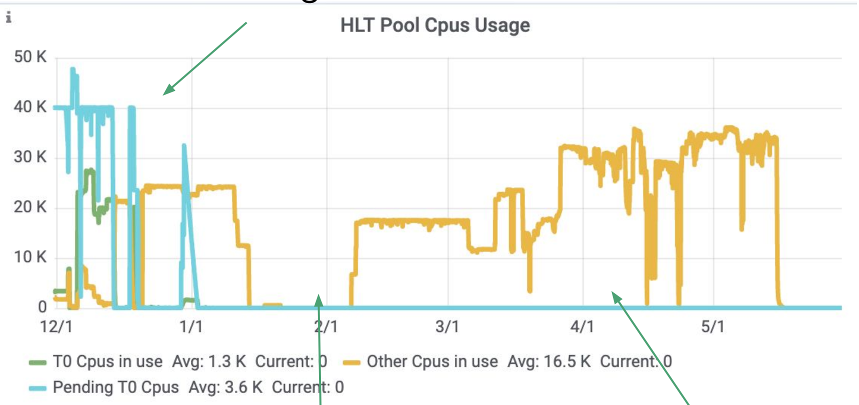

Parking B: started with 15% done


HI re-reco: 50% done


MC18: 13 B done

Status of resources

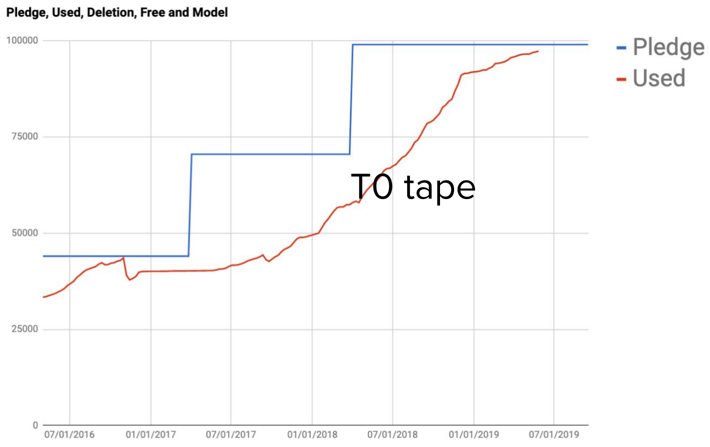
- **Full utilization of CPU resources, including HLT when available**
- **Storage situation under control**
- **Tape:**
  - CERN is full, as expected, with no more increases expected before 2021
  - T1s are slowly increasing, big utilization will come as soon as UL starts
- **Disk:**
  - Maintained by Dynamo @ 90% as expected. “Unmovable” stuff @ 70%

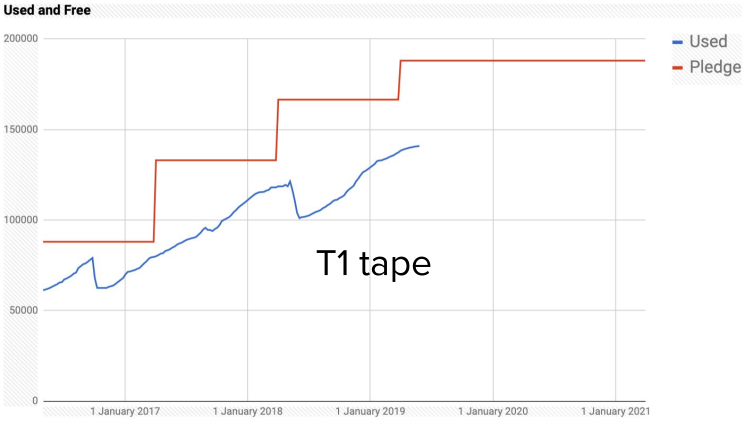


T0 running

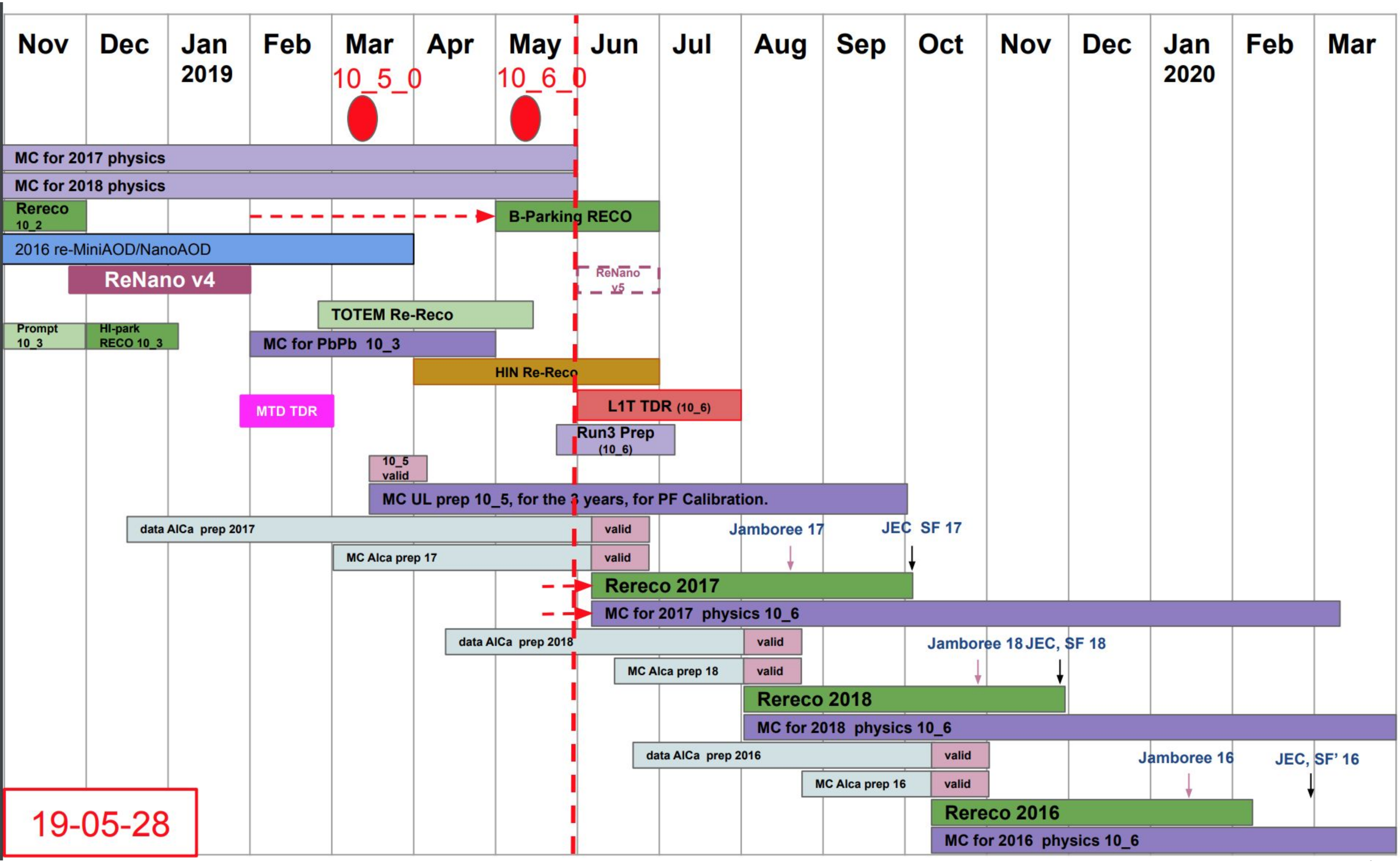
HLT off (power/cooling not available)

HLT available





Production Plan for 2019



Nov Dec Jan 2019 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec Jan 2020 Feb Mar

10_5_0

10_6_0

MC for 2017 physics

MC for 2018 physics

Rereco 10_2

B-Parking RECO

2016 re-MiniAOD/NanoAOD

ReNano v4

ReNano v5

TOTEM Re-Reco

Prompt 10_3

HI-park RECO 10_3

MC for PbPb 10_3

HIN Re-Reco

MTD TDR

L1T TDR (10_6)

Run3 Prep (10_6)

10_5 valid

MC UL prep 10_5, for the 3 years, for PF Calibration.

data AlCa prep 2017

valid

Jamboree 17

JEC SF 17

MC Alca prep 17

valid

Rereco 2017

MC for 2017 physics 10_6

data AlCa prep 2018

valid

Jamboree 18

JEC, SF 18

MC Alca prep 18

valid

Rereco 2018

MC for 2018 physics 10_6

data AlCa prep 2016

valid

Jamboree 16

JEC, SF' 16

MC Alca prep 16

valid

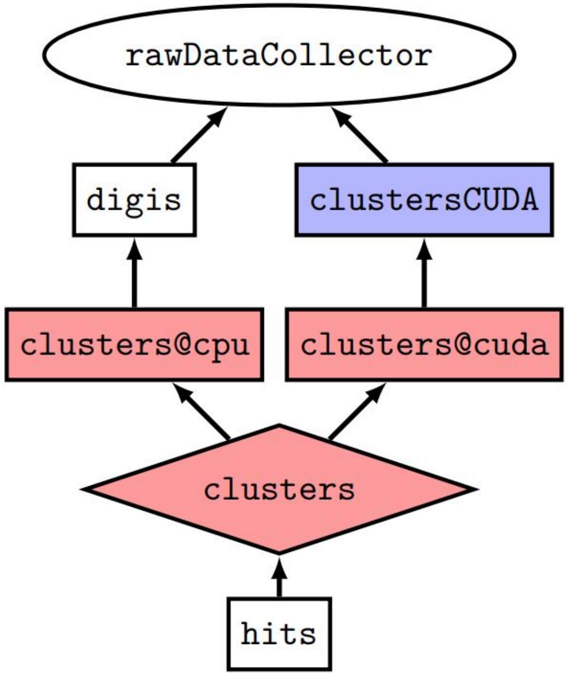
Rereco 2016

MC for 2016 physics 10_6

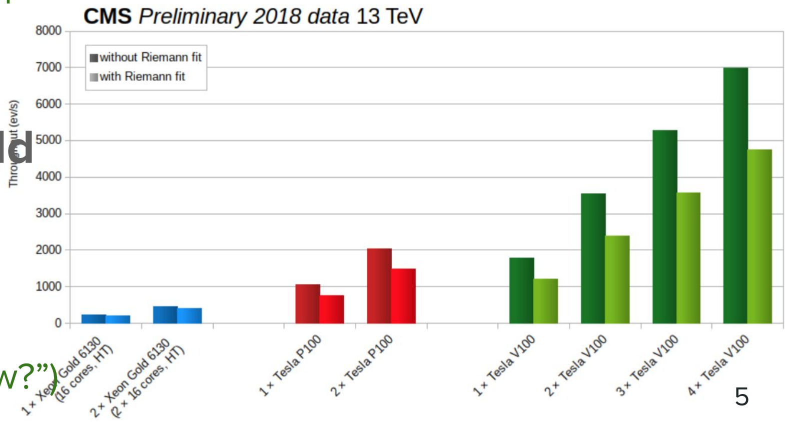
19-05-28

Some notable developments and news: Heterogeneous frameworks

- **As presented @ Feb LHCC, CMS has been inserting in its SW fully general Heterogeneous capabilities:**
  - **"SwitchProducers"**: "equivalent" modules or module sequences, with the same initial input and producing the same final output. Used to provide portability between different architectures
  - **"ExternalWork"**: Enables TBB to run (other) tasks in CPU while something is being run elsewhere, e.g. in accelerator(s) -- so **CPU is not waiting for GPU to finish**, for example
    - **acquire()** ➔ prepares and send work to an accelerator; CPU is then freed
    - **produce()** ➔ called when GPU is done, finalizes outputs
  - **Full tracking**: module by module, it is possible to access which specific version was used
- **Late decision when needed: CMSSW can be told to use version A or B:**
    - At configuration level
    - At autodiscovery level (when starting on a node)
    - Module by Module ("which is the free resource now?")

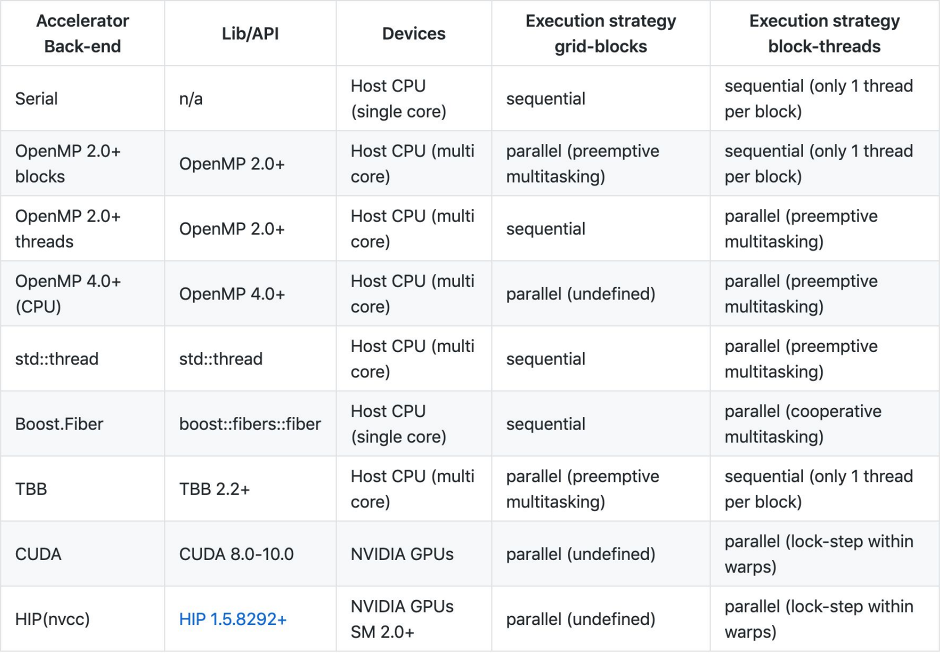


Patatrack Demonstrator

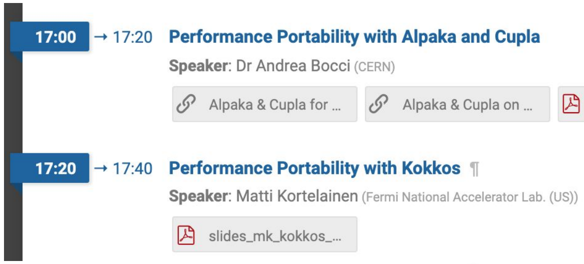


Next steps ...

- We can seamlessly run two codes (CPU and GPU, CPU and FPGA, ...) for a single task, but at the moment we still need to write them
    - 2x (Nx?) the devel effort
    - Physics validation problematic
- Next step: try and find a way to have (efficient) GPU and CPU (and eventually FPGA, AMD, ...) from a single code base
- ECoM2X initiated studies to evaluate **Alpaka** and **Kokkos**
    - Initial evaluation very positive
    - No performance loss wrt direct CUDA
    - Decent performance on CPU, and possibility to use TBB
    - Very easy to port existing CUDA to them
        - For the time being, we can continue using CUDA directly



Accelerator Back-end	Lib/API	Devices	Execution strategy grid-blocks	Execution strategy block-threads
Serial	n/a	Host CPU (single core)	sequential	sequential (only 1 thread per block)
OpenMP 2.0+ blocks	OpenMP 2.0+	Host CPU (multi core)	parallel (preemptive multitasking)	sequential (only 1 thread per block)
OpenMP 2.0+ threads	OpenMP 2.0+	Host CPU (multi core)	sequential	parallel (preemptive multitasking)
OpenMP 4.0+ (CPU)	OpenMP 4.0+	Host CPU (multi core)	parallel (undefined)	parallel (preemptive multitasking)
std::thread	std::thread	Host CPU (multi core)	sequential	parallel (preemptive multitasking)
Boost.Fiber	boost::fibers::fiber	Host CPU (single core)	sequential	parallel (cooperative multitasking)
TBB	TBB 2.2+	Host CPU (multi core)	parallel (preemptive multitasking)	sequential (only 1 thread per block)
CUDA	CUDA 8.0-10.0	NVIDIA GPUs	parallel (undefined)	parallel (lock-step within warps)
HIP(nvcc)	HIP 1.5.8292+	NVIDIA GPUs SM 2.0+	parallel (undefined)	parallel (lock-step within warps)



17:00 → 17:20 **Performance Portability with Alpaka and Cupla**

Speaker: Dr Andrea Bocci (CERN)

17:20 → 17:40 **Performance Portability with Kokkos** ¶

Speaker: Matti Kortelainen (Fermi National Accelerator Lab. (US))

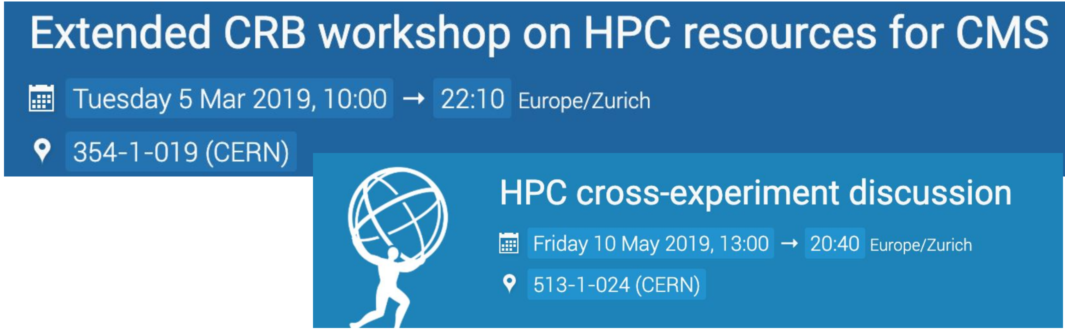
preliminary!

- CUDA version
    - `Output: 1699 modules in 3232.7 us`
- Cupla version with CUDA backend
    - `Output: 1699 modules in 3238.92 us`
- naive CPU version
    - `Output: 1699 modules in 2227.16 us`
- Cupla version with serial CPU backend (synchronous)
    - `Output: 1699 modules in 2218.01 us`
- Cupla version with parallel TBB backend (asynchronous)
    - `Output: 1699 modules in 816.46 us`

No loss...

No loss...

Discussion on HPCs



Extended CRB workshop on HPC resources for CMS

Tuesday 5 Mar 2019, 10:00 → 22:10 Europe/Zurich

354-1-019 (CERN)

HPC cross-experiment discussion

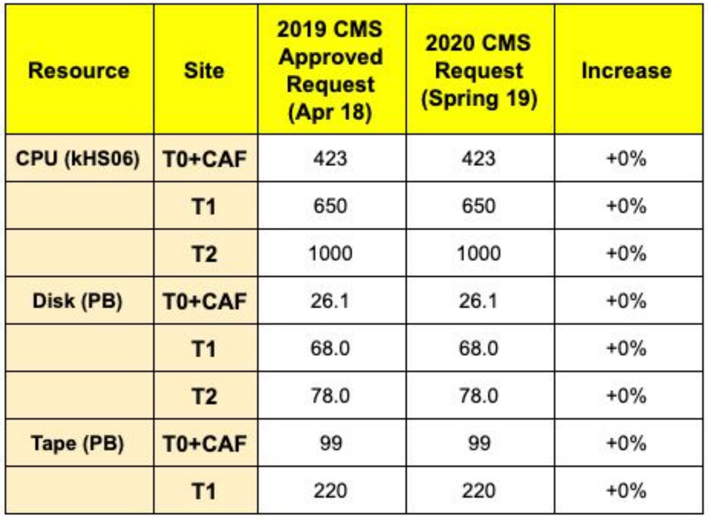
Friday 10 May 2019, 13:00 → 20:40 Europe/Zurich

513-1-024 (CERN)

- **Disclaimer: our life would be easier w/o the need to use HPC systems… But if we need to …**
- **We had a one-day meeting with contacts from our FAs to assess their plans, their resources**
- **Many messages collected, a few highlights:**
  - Operationally, we cannot have CMS operation teams handling the complexity of each site; this must be hidden behind some layers which are best handled by closeby communities (HepCloud is a clear example)
    - We just opened a L3 position for a liason with such centers
  - Contacts with HPC centers are easier for people in the same region / funding agencies. CMS will offer central support, but at the same time we encourage our CMS colleagues to try and open direct communication channels
- **Then, ATLAS+CMS+LHCb meeting on HPC**
  - A lot of understanding on our colleagues solutions / ideas
  - Attempt to investigate common / synchronized grants
- **CMS documents on HPC made public to WLCG:**
  - A technical handshaking document on CMS needs and solutions
  - An Exec Summary on our motivations and ideas on HPC

Update on resource requests for 2020

- CMS has confirms a request of +0% (wrt 19) in all the areas
- → **approved at the April RRB**
- CMS planned activities in 2020:
  - Tails of the ultra-legacy processing
  - Analyses as in 2018
  - Run III preparation, including a dress rehearsal of the new sw components
    - Rucio, DD4HEP, CRIC, …
- For 2021, CMS updated its modelling in view of recent LHC planning; with respect to 2019 the up-to-date figures are
  - CPU: +30%
  - Disk: +20%
  - Tape: +25%
  - (all to be confirmed in Fall’19) - **used to be a flat +30%**
- For October RRB, we are re-evaluating 2021 requests, hoping to grasp a more solid view of the LHC running conditions



Resource	Site	2019 CMS Approved Request (Apr 18)	2020 CMS Request (Spring 19)	Increase
CPU (kHS06)	T0+CAF	423	423	+0%
	T1	650	650	+0%
	T2	1000	1000	+0%
Disk (PB)	T0+CAF	26.1	26.1	+0%
	T1	68.0	68.0	+0%
	T2	78.0	78.0	+0%
Tape (PB)	T0+CAF	99	99	+0%
	T1	220	220	+0%

Table 5: *CMS resource request for 2020. The first column of numbers shows CRSG Spring’18 recommendations for 2019 resources, the second shows the 2020 CMS request, while the third shows the relative increase with respect to the 2020 CMS request (which is unchanged since the CMS Fall’18 request).*

Conclusions

- **From the operational point of view, CMS O+C has been able to deliver what promised. Some delays, due to conditions external to O+C (calibrations, extended time for algo devel) are typical and not problematic**
- **We have a sound program of work for the rest of the year (and indeed at least to the end of Spring 2020), which includes mostly UL production and L1 TDR preparation**
- **We are progressing in our understanding on how to better handle heterogeneous computing**
    - Not strictly needed for Run III, more a long term investment
- **We will update**
    - Phase-2 projections: at September LHCC
    - Run III requests: at October RRB