

subMIT as analysis facility

Mariarosaria D'Alfonso

Massachusetts Institute of Technology

[member of CMS collaboration at LHC]



Re-thinking the data analysis center

HL-LHC particle physics experiments will record an unprecedented scientific data volume at multi-exabyte scale. The current LHC computing model will not provide the required data processing and storage capabilities even with foreseen hardware evolution. Future experiments with big data - Dune, LSST, Square Kilometer Array (SKA), Electron Ion Collider (EIC) - face the same challenges.

The physical sciences community is actively investing in analysis facilities (AF).

The relentless growth in the volume of data created every day affect all the day of our lifes as well not only our research

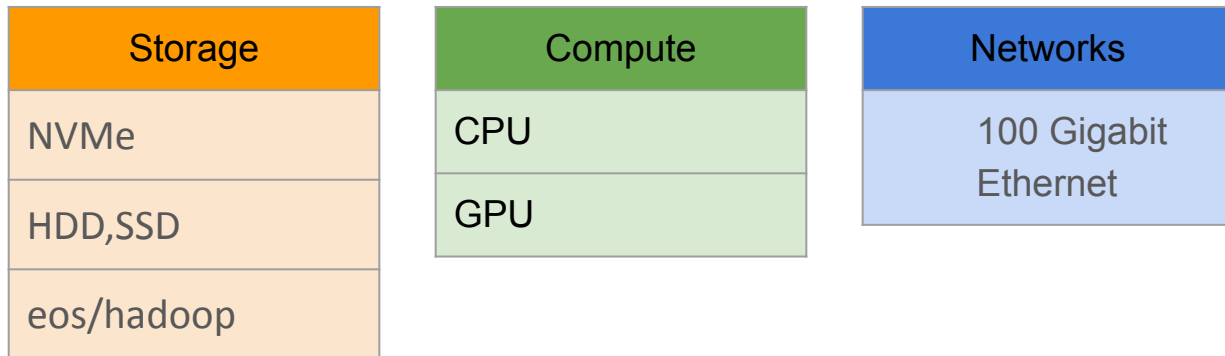
Re-thinking the data center

Once upon a time , Analysis Facility (AF) was login terminals with batch access.

New concept: An Analysis Facility is not just a physical structure- the set-up and design, power, physical security and uptime - but a global, managed ecosystem with the capacity to share resources within and beyond physical boundaries.

Hardware infrastructure

Integrating legacy and new architectures in a single, manageable ecosystem.



Role of the software

An increasing percentage of critical operational and management functions is enabled in the software layer rather than by the underlying hardware.



Algorithm
HEP tools: ROOT, RooFIT
non HEP tools: python-based ecosystem

Tools
machine learning
analytics engines like Dask, Apache spark

Portability of software and reproducibility of environments
conda
singularity or docker containers distributed on CVMFS

Opportunity for customization

Take what you need

- **Scheduling computation with max efficiency:**
 - including options beyond batch jobs+merging output (Slurm, HTcondor, Dask or Apache Spark)
 - Single terminal or jupyter notebook for interactive analysis
- Aiming at **standardised environments** for data analysis as well as in machine learning training
 - local instances as well in batch systems
 - I.e. singularity or docker containers distributed through CVMFS
- Choose among the **disk/computing as it fits**
 - `/scratch/submit/ NVMe disk`  vs `/data/submit/ as gluster volume` vs `/mnt/T2_US_MIT` 

Modular system design with structure robust against changes in the computing environment, so that changes in underlying code can be handled without an entire overhaul of the structure

Requires little knowledge or manpower to get started:

→ undergraduate students: IAP 2022 (Hrere,SUEP), IAP 2023 (FCC)

Outcome-based metrics

The focus should be on optimizing individual hardware infrastructure components extracting all the benefits made possible by software defined technology.

Desiderata: response needs to be immediate and spot-on to meet increasing expectations for **availability, scalability and speed**.

Consider the metrics currently being collected by cities using intelligent transportation systems. These systems provide real-time traffic information and alerts to help drivers avoid congestion and help cities improve roadways to meet citizens' needs. Instead of measuring CPU, memory and disk utilization to assess the systems' success, cities are measuring reduction in traffic, fuel consumption and carbon emissions.

Challenges of HL-LHC analyses

Integrated luminosity $L = 160 \text{ fb}^{-1}$ in Run 2; expected to reach $L > 3000 \text{ fb}^{-1}$ during High-Luminosity LHC (HL-LHC)

New physics opportunities ahead with analysis challenges:

- Higher pile-up (Run5 ~ 200)
- Higher trigger rates
 - Record tailored signatures, going into the tails
- More MC simulated events to match the data luminosity
- More unconventional signatures
- More precisions physics
 - Need the calibration constants follows demands
 - More parametrizations \rightarrow improved/flexible storage

CMS data organization before 2020

Centralised data processing tasks such as simulation or reconstruction.
Most data is written in the ROOT file format.



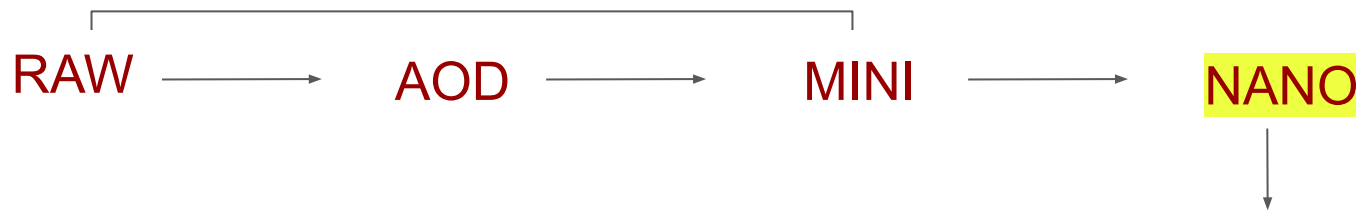
Run1/Run2 analysis dataformat

Data Management: with distributed computing in the form of the grid for the

Data Access: Various protocols were used for direct reads (RFIO, dCap, XRootD, etc.)

CMS data organization

Centralised data processing tasks such as simulation or reconstruction.
Most data is written in the ROOT file format.



lightweight data tier used in the analysis facilities
“fundamental type and arrays thereof” format,
can be read from bare root

→ even non-CMS members can trivially run on *open nanoAOD*.

column-based organization favors large statistical analysis and
prompted the design of the AF.



Big picture on HL_LHC needs (simulation/reconstruction)

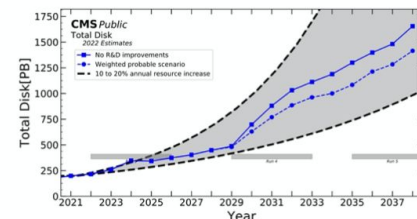
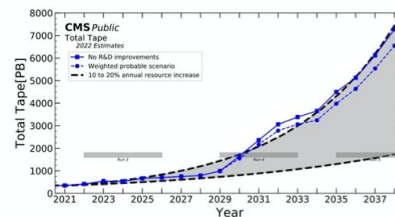
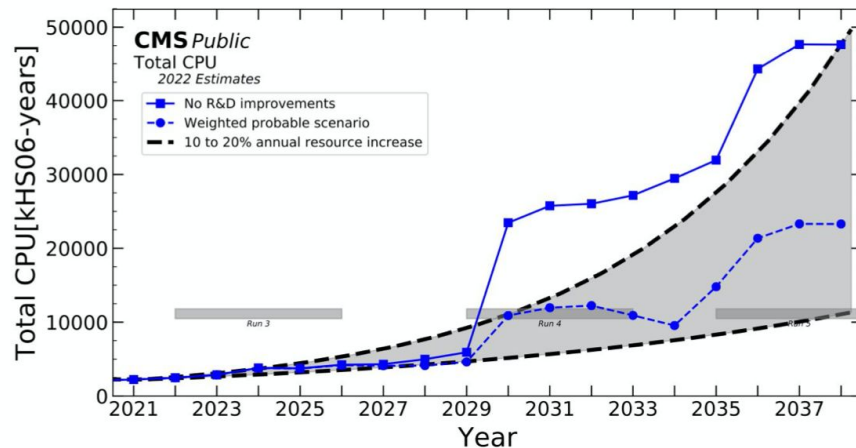
July 2022 Estimates for CPU, Tape, and Disk usage for the next 15 years

- Each plot has band showing a 10% (low) to 20% (high) increase in resources year by year
- Dashed blue lines are (solid) no R&D improvements and (dashed) likely R&D improvements

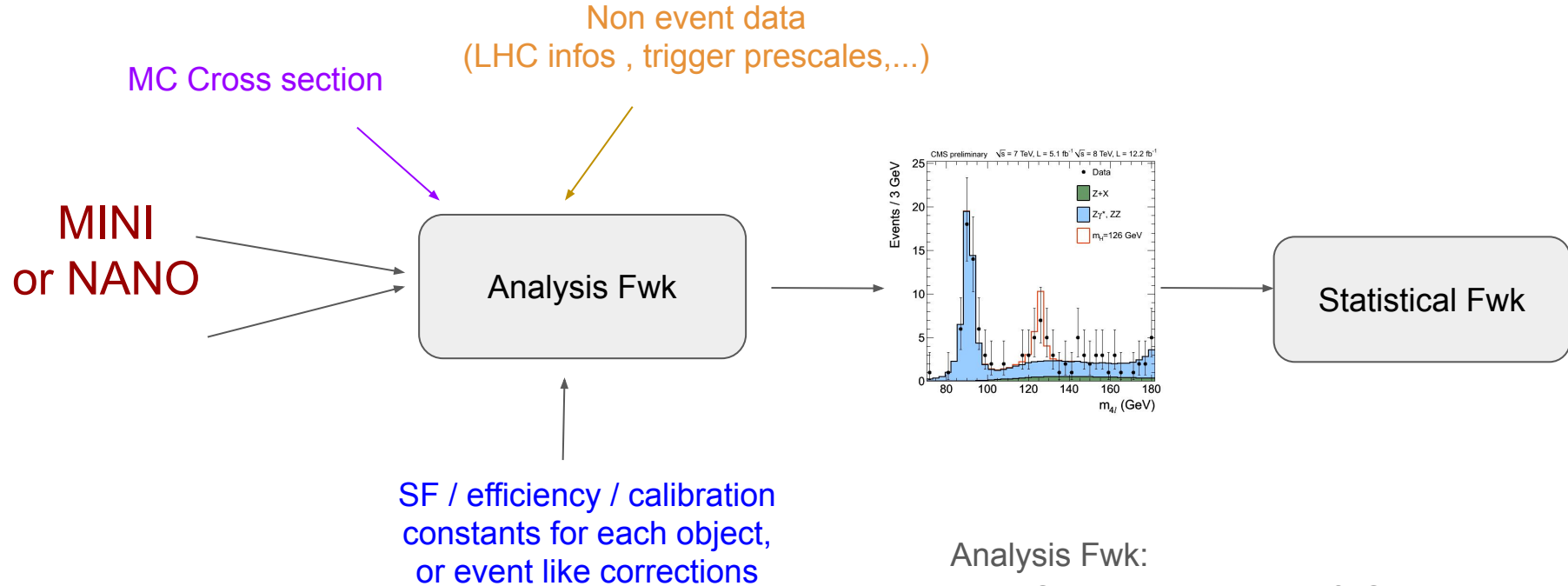
Takeaways:

- CPU use is most critical
- Tape is going to be on the edge
- Disk is probably ok:

Raw format will dominate the disk usage,, adoption of smaller formats (NanoAOD) can have large impact on disk usage



Analysis ingredients

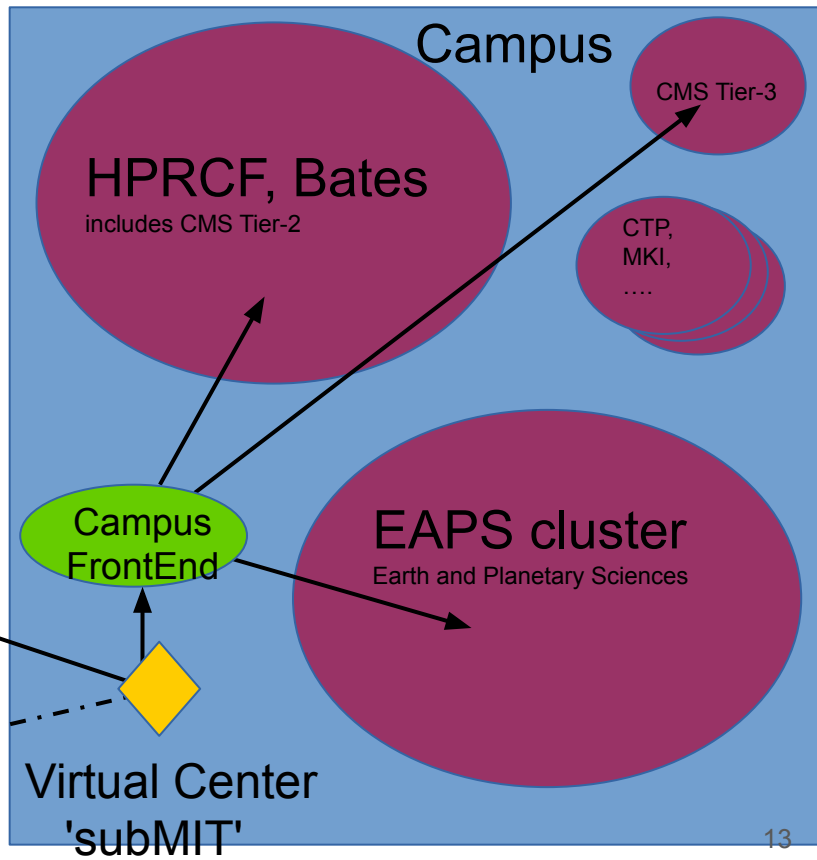
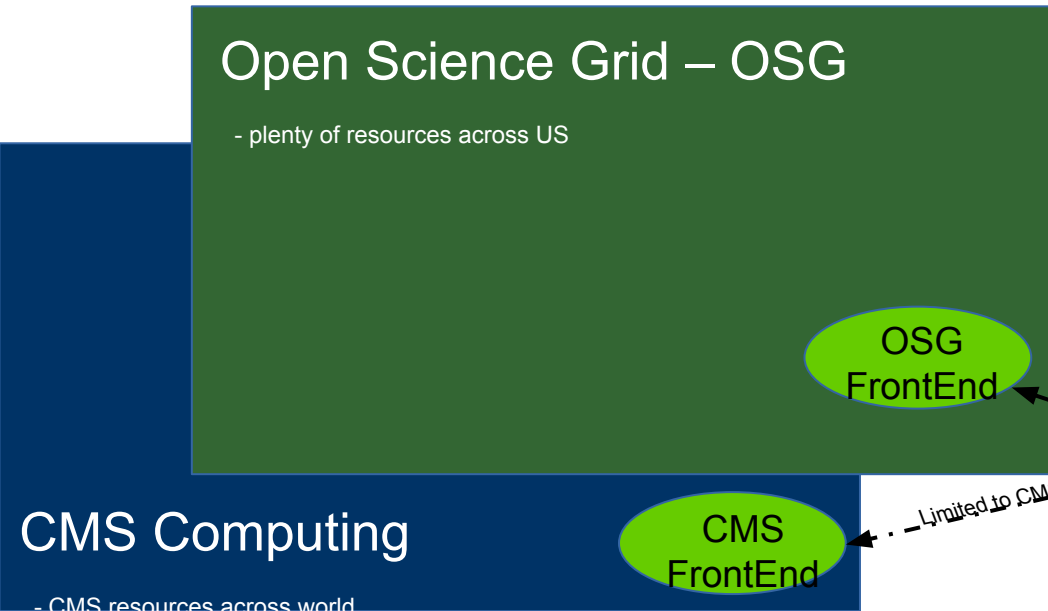


Analysis Fwk:

- Small cpu layer on top of I/O
- Intensive calculation of systematics as input of the fit

CMS connection to subMIT

Connected to all resources on campus



Examples of workflows on subMIT from LHC/CMS

Very different analysis requirement

1. Search for **rare decay of the Higgs Boson**:
 - a. largely profit of event size reduction, simple calculations and almost interactive analysis
 - b. small final dataset for ML inference, GPU used for training
2. Search for **“Soft Unclustered Energy Patterns (SUEPs)”**:
 - a. real time analysis reclustering the “jets”, select SUEP candidates and boost in that frame
 - b. heavily relying in the parallelization (batching HTcondor)
3. Measurement of the **W boson mass**:
 - a. challenge in bookkeeping of templates for systematics variation of uncertainty weights for both background and signal, i.e. build $O(10^3)$ replicas of the final histograms
 - b. need multithreading and memory-based challenges
 - c. need a big machine for now
 - d. GPU used for final fit

Common feature: use the nanoAOD simplified data format as input

HRARE - MODEL



INPUTS

- MINI → NANO within CMSSW Kraken
- SKIMS interactive now should be with batching



other BKGs (100 TB)

`/mnt/T2_US_MIT/hadoop/cms/store/user/paus/nanohr/D01/`

main BKG (GJets is 5TB) on gluster volume

`/data/submit/cms/store/user/mariadlf/nano/D01`



data skims on NVMe disk

`/scratch/submit/cms/mariadlf/Hrare/SKIMS/D01`

Private Signal Nano

`/data/submit/mariadlf/HrareSIG/D01/2018/`

ANALYSIS

RootDataFrame

interactive

external library with
conda:

- root version 6.26
- correction lib

standalone processing

python:

- FITS
- ML Trainings
- Data MC plots

shell terminal (me)

jupyter (Charlotte,

Kevin)

external library with
conda for ML trainings:

- tensorflow,keras,pandas,numpy

external library with
CMSSW for fits:

- combine

STORAGE

`/work/submit/mariadlf`

5 GB is for conda

30 GB is ntuples and histo

Where to go:

Need get more experience with:

1. data delivery optimization to processing endpoints defining optimal working point between the local disk and the
2. distributed system dynamic shifting load from server to server or AF to another AF to increase utilization

Summary

Positive experience with the subMIT system:

flexible, user oriented, large array of outstanding hw/sw options.

Run3 CMS data analysis will provide a perfect test environment in view of the HL-LHC phase.

Backup



SUEP - MODEL

INPUTS

- Add in all **PFCands** for clustering
- MINI → NANO with SUEPnano (Offline)
- Scouting data using private code

QCD and HT data BKGs (~150 TB)

`/mnt/T2_US_MIT/hadoop/cms/store/user/paus/nanosu/A01/`

Flat Scouting NTuples also stored (~25 TB)

`/mnt/T2_US_MIT/hadoop/cms/store/user/paus/nanosu/E04/`

Analysis skims output in data (~100GB)
Stored as hdf5 files with pandas dataframes

`/data/submit/cms/store/user/`

ANALYSIS

Columnar analysis with Coffea

Analysis run through HTCondor

1. CMS global pool
2. MIT T2 and T3

Accesses files via xrootd

Uses Coffea Singularity
(Through CVMFS)

Packages added to Coffea docker
specifically for SUEP analysis:

1. fastjet (Awkward input)
2. onnx
3. pytorch
4. pytables

standalone processing

hdf5 → pickle and root files
(~3-5 GB)

Merged and analyzed for (~minutes)

- Histograms
- Dacards/Combine
- Limit plots

Analyzed on submit using
multithreaded code
(All in python3)

Jupyterhub notebooks for plots
analyzing histograms, limits,
systematics, etc

MW - MODEL

- Dedicated large machine at CERN through CMG group/CERN IT
 - 1TB RAM, dual EPYC 128 cores/256 threads total
 - 16x3.84TB NVMe Gen4 storage array (~100Gbytes/sec sequential read)
 - 100gbps NIC (fast access to CERN eos)
- Custom NanoAOD on local SSD array
- Analysis with RDF
 - Special feature: Very Large multidimensional histograms: Use Boost histograms with shared atomic storage to avoid one histogram copy per thread
 - multithreading is a hard requirement for memory reasons
 - Additional challenges for multi-node parallelization (support custom helpers in distRDF and multithreaded dask or spark tasks)
- Software environment: singularity image on CVMFS
- Port to subMIT
 - On large single node to start, with NanoAOD on /data or /scratch
 - Commission/develop needed features in distRDF to enable multi-node parallelization