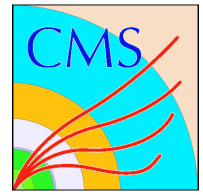




# Towards a Data Lake for the HL-LHC Era

Frank Würthwein

SDSC/UCSD

DOMA Access

November 26th 2019

# Recap from previous presentations

# Start with Data Formats and their expected use

| Data Tier | Data |
|---|---|
| RAW [MB] | 7.4 |
| AOD [MB] | 2.0 |
| MiniAOD [kB] | 200 |
| NanoAOD [kB] | 4 |

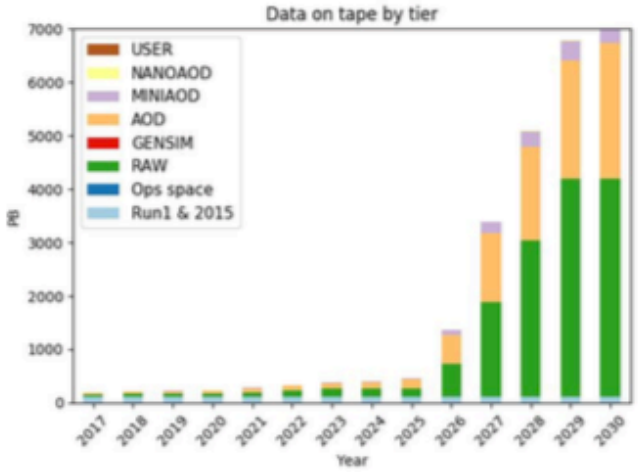
## Primary Processing:
## RAW -> AOD -> Mini -> Nano

## Processing Assumptions:

- ~~All~~ Only Some events are prompt reconstructed
- 25% of events are re-reconstructed (eg for startup)
- There is a reprocessing each year of the current years data
- MC is always made starting from scratch (eg, GEN-SIM is redone)
- In shutdown years, all events in the last 3 years are reprocessed and corresponding MC remade
  - Take 2 years to do this reprocessing as it doesn't fit into 1 year without a resource bump (first shutdown is 2030..)

**Data formats span x1000 in size per event.**
**Files in large data formats are touched at most twice a year.**

# May 2018 Tape Estimate




Use Tape estimates as guidance of size of the total available data.

Dominated by RAW and AOD

Another way of looking at it:

80+160 Billion events/year (Data+MC) = 240B events/year

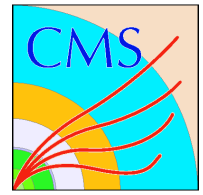
⇒ 7.4MB x 8e10 ~ 6e11 MB ~ 0.5 Exabytes/year of RAW

⇒ 2.0MB x 2.4e11 ~ 5e11 MB ~ 0.5 Exabytes/year of AOD

⇒ 0.2MB x 2.4e11 ~ 0.5e11 MB ~ 50 Petabytes/year of Mini

⇒ 0.004MB x 2.4e11 ~ 0.01e11 MB ~ 1 Petabyte/year of Nano

**The data that is accessed 1-2 times per year is x1000 larger than the data that dominates data analysis use !!!**


# What do we mean by “Data Lake”

Example CMS, but ought to be applicable also to ATLAS

# “Hierarchical Storage”

- Keep most data in “active archive” on cheap, and high latency media (e.g. Tape).
- Keep a “golden copy” on redundant high availability disk.
  - Defines the working set allowed to be accessed.
  - Jobs requesting data not in working set will queue up until data is recalled from archive.
- Regional Caches at processing centers
  - Size of region determined by latency tolerance of application.
  - Cost trade-off between cache size vs network use

# Distances in EU

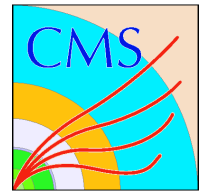

Good goal to set for IO stack to be sufficiently latency tolerant to lose less than 10% in CPU time for access distances of 500-1000 Miles.

# Differences to Today

- There are x5 fewer storage systems to maintain globally.
- There could be only 1 golden copy that requires high reliability.
  - Today all sites are required to be high reliability.
- All caches are JBODs only. No redundancy required.
- Caches can be much more aggressively cleaned.

**Goal: x4 less disk space for better availability of data.**

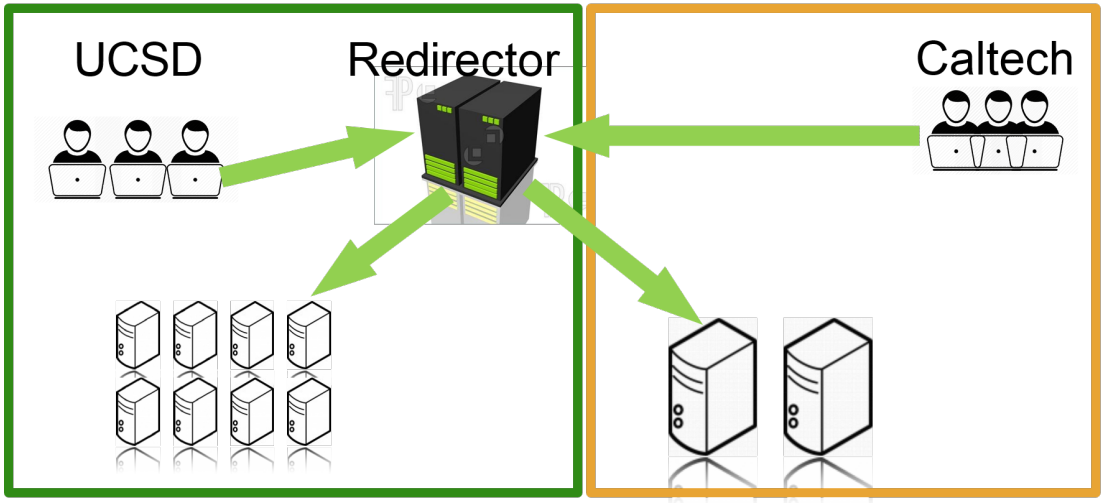
This is ongoing R&D and whether this goal can be achieved remains to be seen.


# Production Scale Caching Prototype

Caltech & UCSD operate a joint PB disk cache.

# Southern California (SoCal)


**CPU in both places can access storage in both places.**

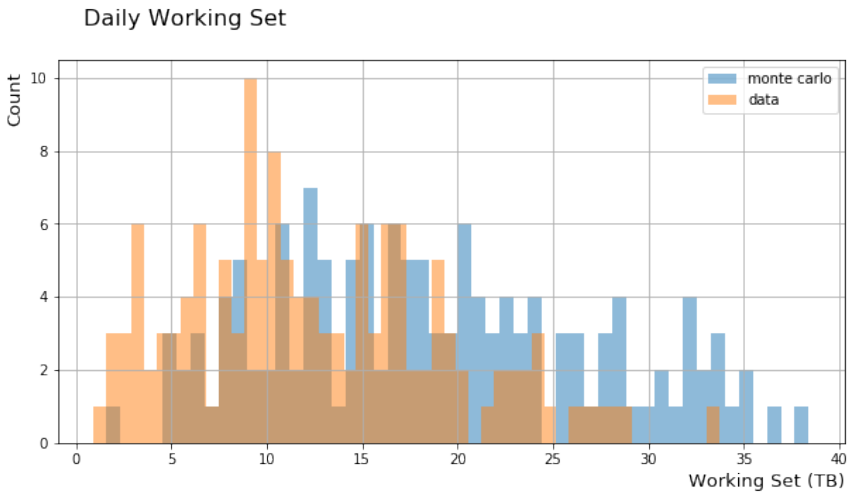
**How much disk space is enough?**

**Cache MINI and measure working set accessed.**

# Working Set (WS) Definitions

WS = sum of sizes of all files accessed in a time period.

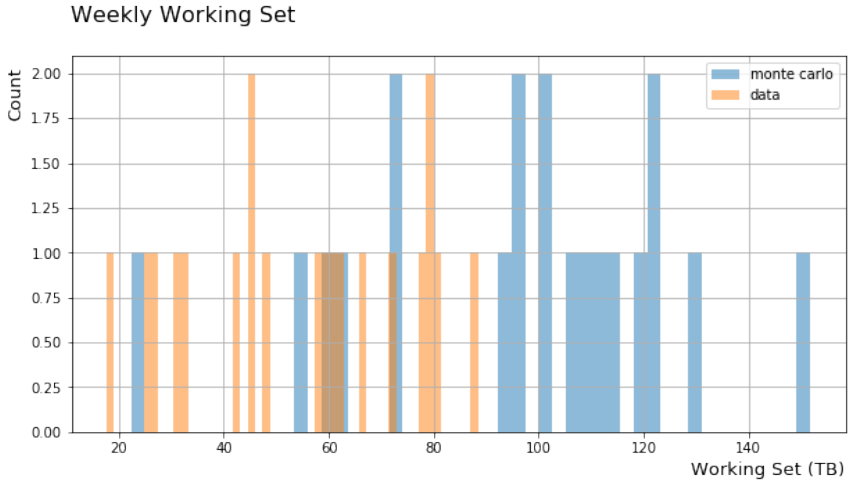
For SoCal cache prototype we measured:

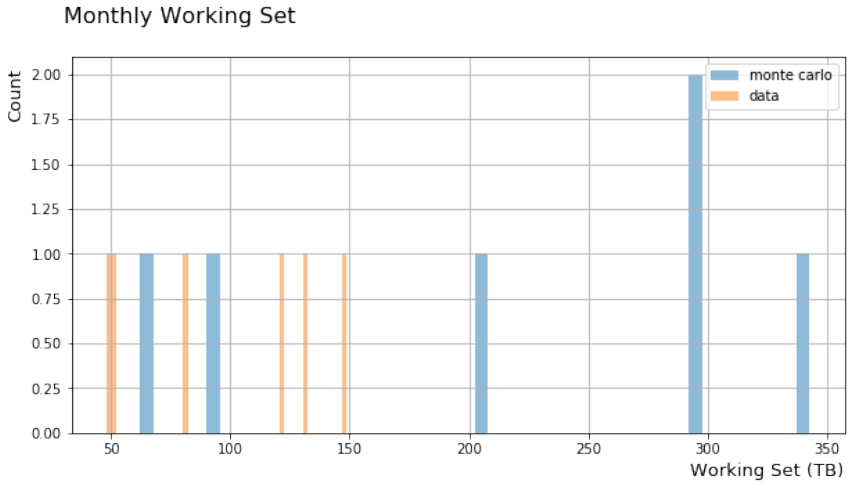



**Few tens TB daily**
**Few hundreds TB monthly**

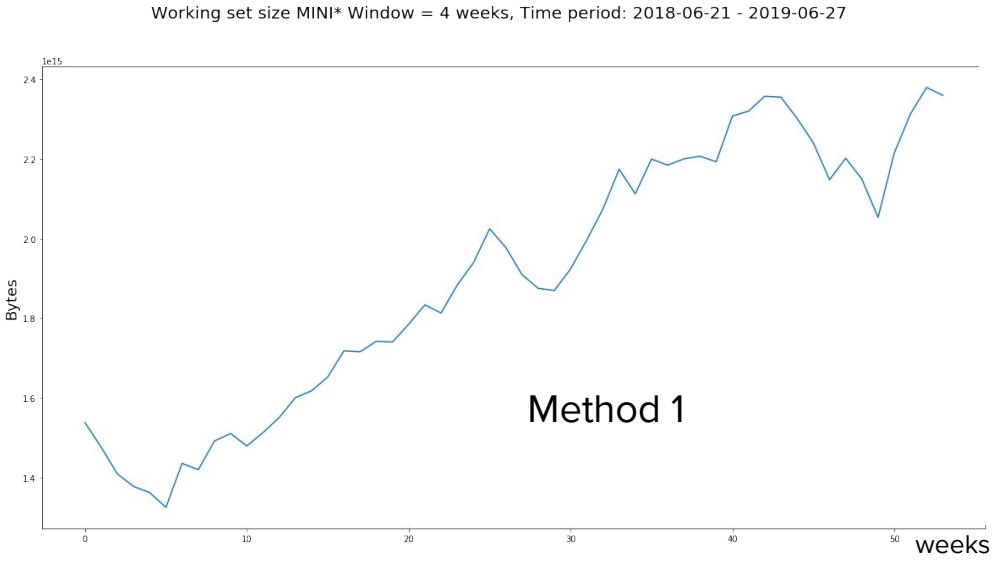
Showing the obvious trade-off between space and network use.

SoCal WS for 10/19 = 451 TB



# And what about globally?




Method 1:

1. Look at the unique MINI* **data-sets** accessed globally (at all sites) within a four week window and calculate their size.
2. Move the window 1 week at a time for a year worth of data from the Global pool ClassAds
3. Results: The **monthly working set** is somewhere between 1.6PB and 2.4 PB

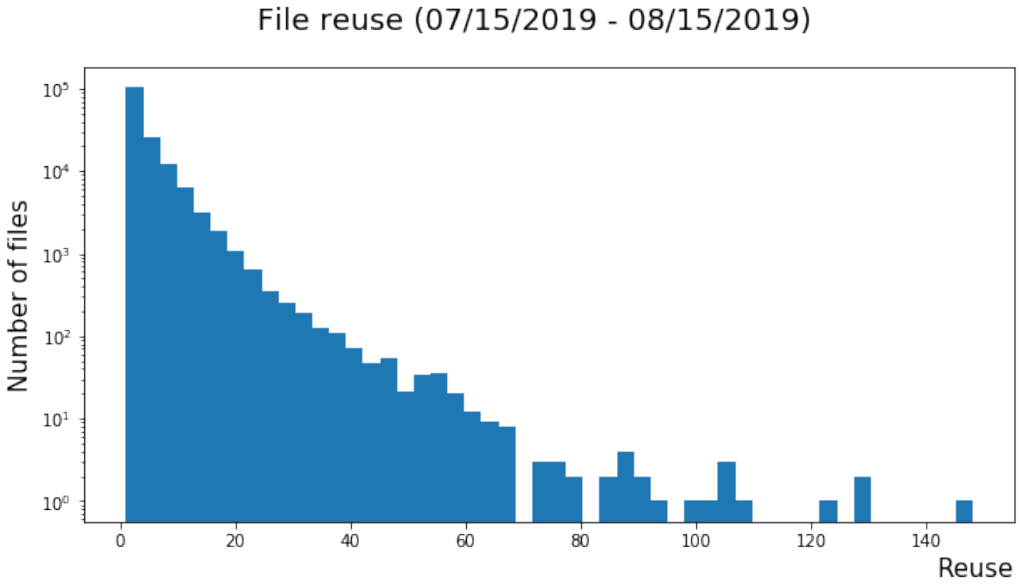
Monthly global working set grew from 1.6PB to 2.4PB in a year

Total available data in MINI* is now 7.8PB

**Even a large set of sites like SoCal (~20k cores) will see only ~1/5$^{th}$ of the global WS accessed in a typical month.**

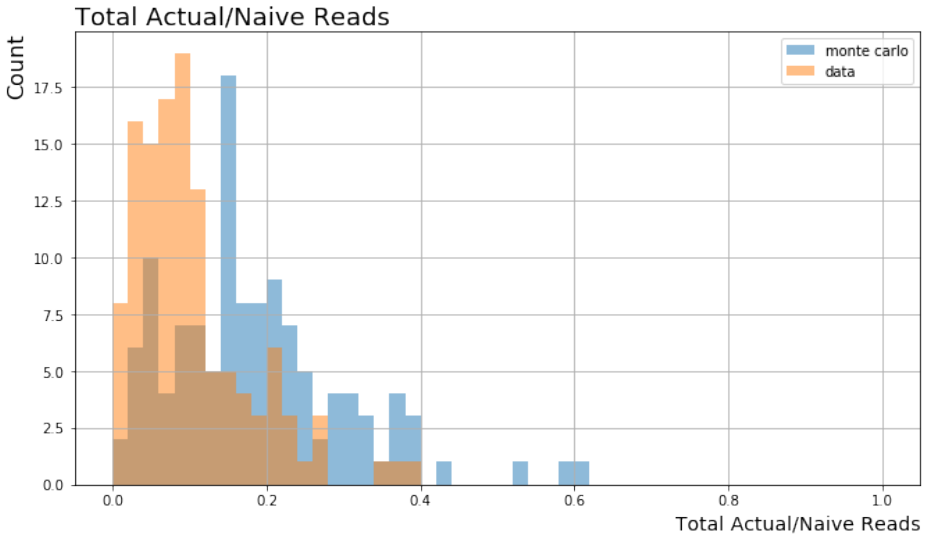
# Next R&D Steps

# A closer look …




**Most likely reuse of files in cache is zero**

Prediction of reuse could lead to significantly better cache use
=> less disk space needed




**For most files, less than 20% of file is read.**

Prediction of partial file reads could lead to significantly better cache use
=> Less disk space needed

# A closer look …

**Most likely reuse of files in cache is zero**

R&D being pursued by Diego Ciangottini, Daniele Spiga, et al.

<=

Prediction of reuse could lead to significantly better cache use
=> less disk space needed

R&D not presently done by anybody

<=

**For most files, less than 20% of file is read.**

Prediction of partial file reads could lead to significantly better cache use
=> Less disk space needed

(might be important for Analysis Systems ServiceX applicability)

# Summary & Conclusions

- Caltech & UCSD are operating a petabyte scale cache that is accessed from CPU at both locations.
- Starting to understand data reuse … but a lot more work should be done.
  - Lots of possibilities to reduce disk space needed in cache by improved caching algorithms. (presently use LRU)
  - Possibilities for further storage needs reduction via partial file caching.