ALICE

Data Formats and Impact on Federated Access

Costin.Grigoras@cern.ch
Andrei.Gheata@cern.ch

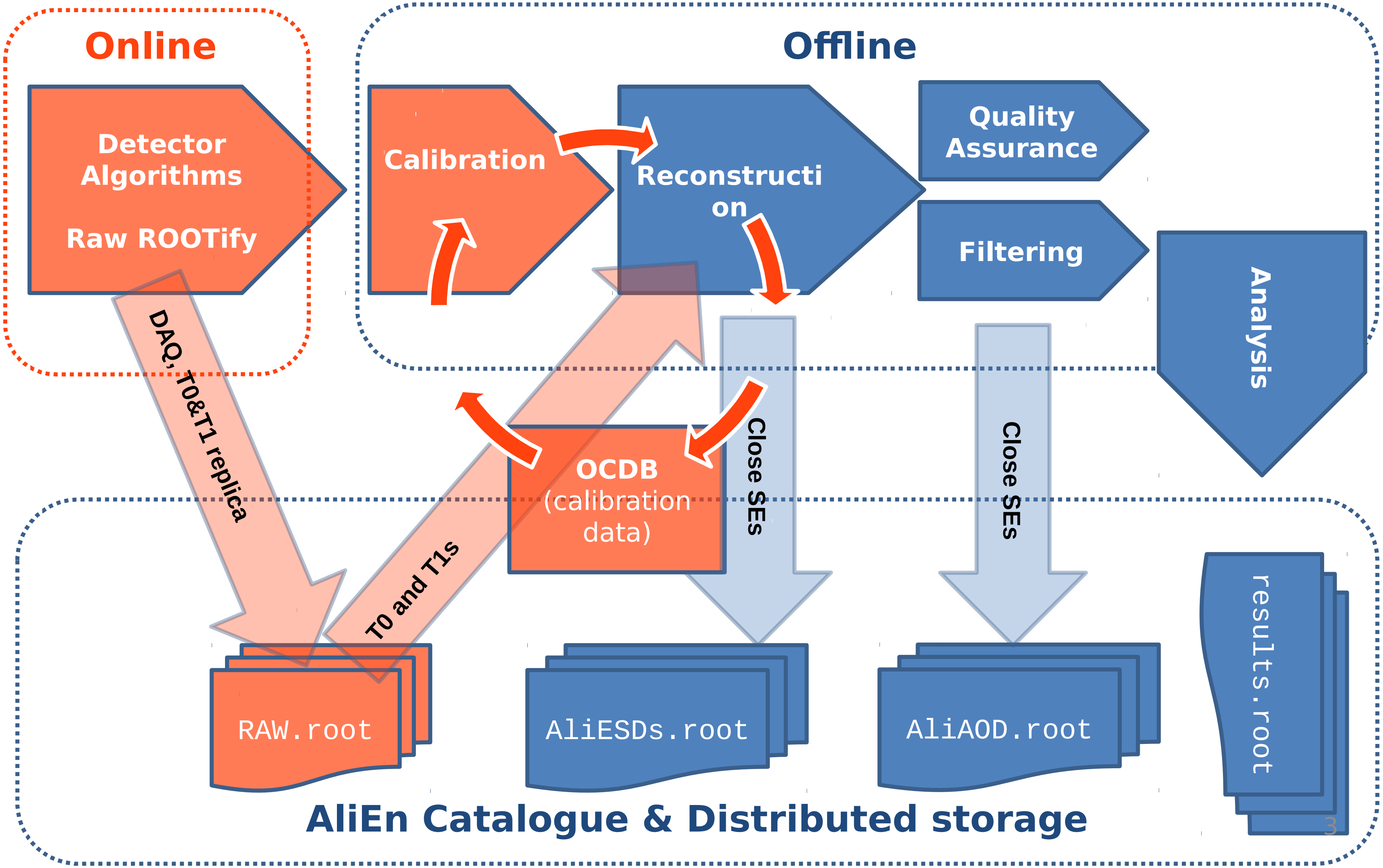
General points

- All data files are in ROOT format
  - Raw data is “ROOTified” at DAQ level
  - Reconstructed & MC data (ESDs)
  - Analysis input data (AODs)
  - Intermediate and merged analysis results, QA, etc.
- Complex detector with 18 subsystems
  - Large event size dominated by TPC contribution
    - Up to 5-8 MB / event (uncompressed) for Pb-Pb
- Data is accessed directly from storage with the Xrootd protocol

ALICE data flow

Raw data only

Raw and MC data



Main containers

- ESDs: 790 branches
- AODs: 400 branches
  - Most analysis can run on AODs
  - Some need the extra details in ESDs
- Formats highly dependent on AliRoot types, deep object hierarchy

Event size by collision type

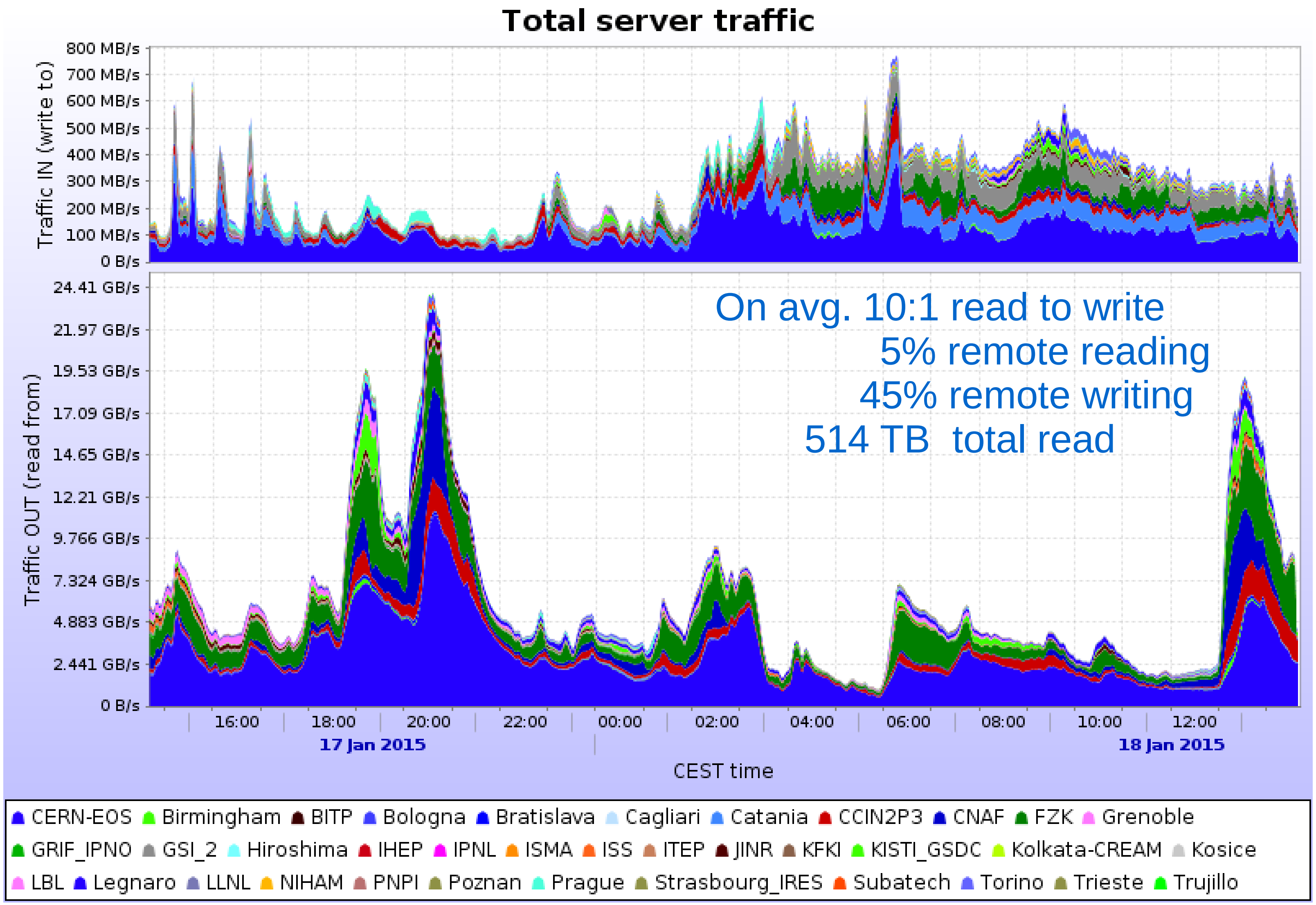
Collision	Year	RAW size [kB/ev]	ESD size (*) [kB/ev]	Reco time seconds/ev	Bandwidth kB/s	AOD size (*) [kB/ev]
p-p	2010	550	62	6.2	89	10.5
	2011	500	61	4.8	104	6.7
	2012	1820	113	7.5	243	9.6
Pb-Pb	2010	11380	1710	52	219	365
	2011	5490	4070	132	42	1800
p-Pb	2012	612	271	6.5	94	60
	2013	1660	1640	43.7	38	379

(*) Compressed data with compression factor ~4-5

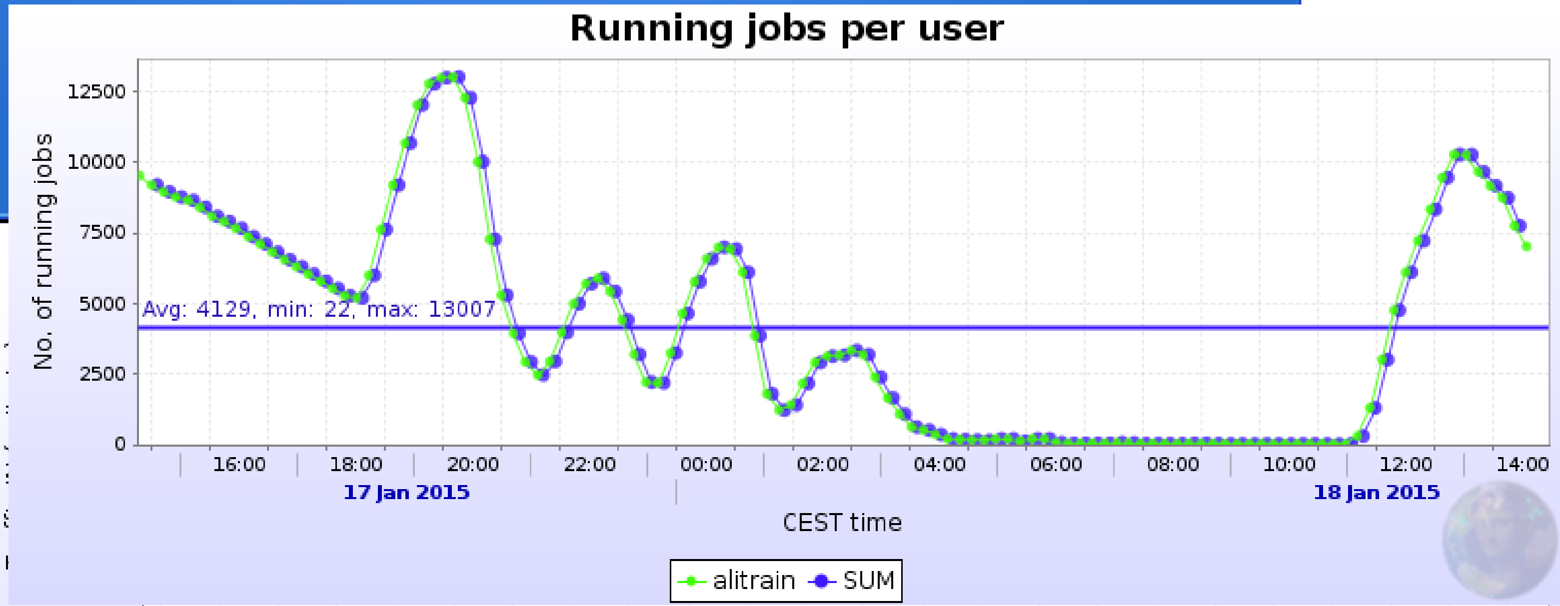
I/O contributors

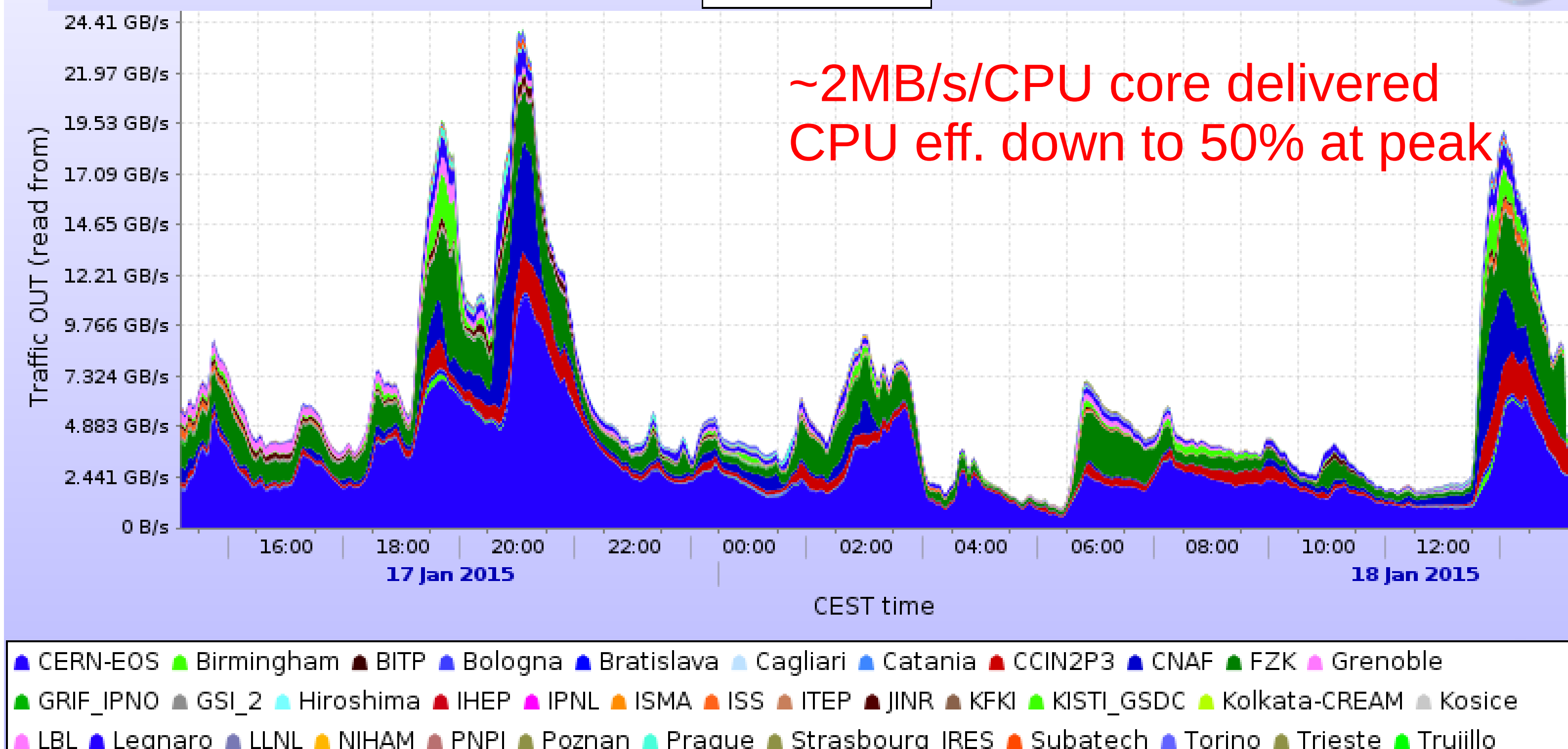
- Raw data reconstruction: read from SE, reconstruct (=>ESDs) and filter (=>AODs), upload them
  - 2-3 data passes
- Simulation: ESDs and AODs at low throughput
- Re-filtering: AODs from ESDs (infrequent)
- Analysis: I/O bound reading of ESDs or AODs
  - Largest contributor to storage I/O

Data rates



Total server traffic
Traffic IN (write to)
800 MB/s
700 MB/s
600 MB/s
500 MB/s
400 MB/s
300 MB/s
200 MB/s
100 MB/s
0 B/s
Traffic OUT (read from)
24.41 GB/s
21.97 GB/s
19.53 GB/s
17.09 GB/s
14.65 GB/s
12.21 GB/s
9.766 GB/s
7.324 GB/s
4.883 GB/s
2.441 GB/s
0 B/s
16:00 18:00 20:00 22:00 00:00 02:00 04:00 06:00 08:00 10:00 12:00
17 Jan 2015
18 Jan 2015
CEST time
On avg. 10:1 read to write
5% remote reading
45% remote writing
514 TB total read
CERN-EOS Birmingham BITP Bologna Bratislava Cagliari Catania CCIN2P3 CNAF FZK Grenoble
GRIF_IPNO GSI_2 Hiroshima IHEP IPNL ISMA ISS ITEP JINR KFKI KISTI_GSDC Kolkata-CREAM Kosice
LBL Legnaro LLNL NIHAM PNPI Poznan Prague Strasbourg_IRES Subatech Torino Trieste Trujillo


Running jobs per user
No. of running jobs
12500
10000
7500
5000
2500
0
Avg: 4129, min: 22, max: 13007
16:00 18:00 20:00 22:00 00:00 02:00 04:00 06:00 08:00 10:00 12:00 14:00
17 Jan 2015
18 Jan 2015
CEST time
alitrain
SUM


~2MB/s/CPU core delivered
CPU eff. down to 50% at peak
Traffic OUT (read from)
24.41 GB/s
21.97 GB/s
19.53 GB/s
17.09 GB/s
14.65 GB/s
12.21 GB/s
9.766 GB/s
7.324 GB/s
4.883 GB/s
2.441 GB/s
0 B/s
16:00 18:00 20:00 22:00 00:00 02:00 04:00 06:00 08:00 10:00 12:00
17 Jan 2015
18 Jan 2015
CEST time
CERN-EOS Birmingham BITP Bologna Bratislava Cagliari Catania CCIN2P3 CNAF FZK Grenoble
GRIF_IPNO GSI_2 Hiroshima IHEP IPNL ISMA ISS ITEP JINR KFKI KISTI_GSDC Kolkata-CREAM Kosice
LBL Legnaro LLNL NIHAM PNPI Poznan Prague Strasbourg_IRES Subatech Torino Trieste Trujillo

Analysis sample 1

Site	Job eff.	All files	Local files	Remote files	CERN EOS	FZK SE	CNAF SE	CCIN2P3 SE
CERN 2086 jobs (33.13%)	63.35%	2086 files 1.066 MB/s	2086 (100%) 1.066 MB/s		**2086 (100%) 1.066 MB/s**			
FZK 1092 jobs (17.34%)	54.34%	1092 files 1.46 MB/s	1092 (100%) 1.46 MB/s			**1092 (100%) 1.46 MB/s**		
CNAF 920 jobs (14.61%)	88.85%	920 files 2.306 MB/s	920 (100%) 2.306 MB/s				**920 (100%) 2.306 MB/s**	
CCIN2P3 434 jobs (6.893%)	72.22%	434 files 1.717 MB/s	434 (100%) 1.717 MB/s					**434 (100%) 1.717 MB/s**
PRAGUE 381 jobs (6.051%)	71.76%	381 files 1.907 MB/s	380 (99.74%) 1.916 MB/s	1 (0.262%) 0.712 MB/s	1 (0.262%) 0.712 MB/s			
GRIF_IPNO 343 jobs (5.448%)	86.85%	343 files 2.501 MB/s	343 (100%) 2.501 MB/s					
LEGNARO 210 jobs (3.335%)	74.17%	210 files 1.954 MB/s	210 (100%) 1.954 MB/s					
CLERMONT 155 jobs (2.462%)	83.55%	155 files 2.346 MB/s	155 (100%) 2.346 MB/s					
TOTAL 6296 jobs	67.5%	**6296 files 1.478 MB/s 5.207 TB**	**6260 (99.43%) 1.478 MB/s 5.187 TB**		**2086 (33.32%) 1.066 MB/s 1.773 TB**	**1092 (17.44%) 1.46 MB/s 847 GB**	**920 (14.7%) 2.306 MB/s 829.1 GB**	**434 (6.933%) 1.717 MB/s 366.6 GB**
				36 (0.572%) 1.366 MB/s 19.68 GB	**1 (2.778%) 0.712 MB/s 933.8 MB**			**9 (25%) 1.134 MB/s 8.54 GB**

CPU usage efficiency

67.5% of the wall time
38d 14:24 CPU in total

Analysis sample 2

Site	Job eff.	All files	Local files	Remote files	CERN EOS	FZK SE	CNAF SE
FZK 2304 jobs (20.79%)	40.11%	40239 files 5.442 MB/s	39961 (99.31%) 5.489 MB/s	278 (0.691%) 2.442 MB/s	187 (0.465%) 3.726 MB/s	**39961 (99.31%) 5.489 MB/s**	59 (0.147%) 1.618 MB/s
CERN 2276 jobs (20.53%)	42.29%	38741 files 3.914 MB/s	38736 (99.99%) 3.916 MB/s	5 (0.013%) 0.912 MB/s	**38736 (99.99%) 3.916 MB/s**	1 (0.003%) 3.451 MB/s	
CNAF 807 jobs (7.281%)	83.99%	13833 files 10.09 MB/s	13833 (100%) 10.09 MB/s				**13833 (100%) 10.09 MB/s**
CCIN2P3 636 jobs (5.738%)	55.5%	9256 files 6.285 MB/s	9256 (100%) 6.285 MB/s				
LEGNARO 426 jobs (3.843%)	44.46%	6137 files 6.671 MB/s	6137 (100%) 6.671 MB/s				
CLERMONT 398 jobs (3.591%)	58.03%	5760 files 9.159 MB/s	5367 (93.18%) 11.04 MB/s	393 (6.823%) 2.82 MB/s	117 (2.031%) 4.916 MB/s		
KISTI_GSDC 358 jobs (3.23%)	74.3%	5319 files 10.59 MB/s	5293 (99.51%) 11.13 MB/s	26 (0.489%) 0.722 MB/s	10 (0.188%) 0.722 MB/s		4 (0.075%) 0.807 MB/s
NIKHEF 354 jobs (3.194%)	19.24%	5639 files 3.254 MB/s		5639 (100%) 3.254 MB/s	1935 (34.31%) 4.346 MB/s		619 (10.98%) 3.11 MB/s
TOTAL 11084 jobs	38.67%	**165845 files 4.743 MB/s 101.3 TB**	**156341 (94.27%) 4.883 MB/s 95.67 TB**		**38736 (24.78%) 3.916 MB/s 23.99 TB**	**39961 (25.56%) 5.489 MB/s 24.62 TB**	**13833 (8.848%) 10.09 MB/s 8.677 TB**
				9504 (5.731%) 3.188 MB/s 5.618 TB	**2249 (23.66%) 4.235 MB/s 1.424 TB**	**1 (0.011%) 3.451 MB/s 822.3 MB**	**687 (7.229%) 2.86 MB/s 400.3 GB**

IO Management (read + deserialize)

2m 3s / file
30m 49s / job
89.29% of the wall time
237d 5:33 in total

Considerations

- Remote reading avoided as much as possible
  - As fallback if local replica doesn't work
  - Or to speed up last jobs in an analysis
  - Long term average: 2-3% of the volume
- Previously more permissive but had to limit it for job performance issues
  - Both from CPU and network perspectives
- Even local storage element cannot sustain the throughput rates of analysis jobs

Improvement vectors

- Combine more analysis in trains
  - Well underway
- Restrict the number of branches to the minimum needed
  - Hard to do and with little gain since most of the branches are touched
- Filter to more compact formats (nanoAODs)
  - Inflexible analysis code, hard to control productions
- ROOT prefetching revisited
  - Caching enabled by default, helped a lot, little gain from prefetching
- Flatten the AOD format
  - Reduce part of the ~20% time spent in deserialization

Flat AOD exercise

- Use AODtree->MakeClass() to generate a skeleton, then rework
- Keep all AOD info, but restructure the format

```
Int_t fTracks.fDetPid.fTRDncls -> Int_t *fTracks_fDetPid_fTRDncls;  //[ntracks_]
```

  - More complex cases to support I/O:

```
typedef struct {
  Double32_t       x[10];
} vec10dbl32;
Double32_t fTracks.fPID[10] -> vec10dbl32 *fTracks_fPID; //[ntracks_]
Double32_t *fV0s.fPx //[fNprongs] -> TArrayF *fV0s.fPx; //[nv0s_]
```

- Convert AliAODEvent-> FlatEvent
  - Try to keep the full content AND size
- Write FlatEvent on file
- Compare compression, file size and read speed

Flat AOD results

- Tested on Pb-Pb AOD

```
******************************************************************************
*Tree    :aodTree   : AliAOD tree                                            *
*Entries :     2327 : Total =      2761263710 bytes  File  Size =  660491257 *
*        :          : Tree compression factor =   4.18                       *
******************************************************************************
******************************************************************************
*Tree    :AliAODFlat: Flattened AliAODEvent                                  *
*Entries :     2327 : Total =      2248164303 bytes  File  Size =  385263726 *
*        :          : Tree compression factor =   5.84                       *
******************************************************************************
```

- Smaller data (no TObject overhead, TRef->int)
  - 30% better compression
- Reading speed
  - CPU time= 103s , Real time=120s
  - CPU time= 54s , Real time= 64s

Implications

- User analysis more simple, working mostly with basic types (besides the event)
  - Simplified access to data, highly reducing number of (virtual) calls
  - ROOT-only analysis
- Backward incompatible, but migration from old to new format possible
  - Sacrificing performance as first step (built-in transient event converter)
- Much better vectorizable track loops
- This approach is now considered for Run3 and even Run2
- But would not improve the performance of I/O bound jobs

Analysis trains

- Centrally organized user analysis
  - By Physics Working Groups
- Individual user tasks committed to AliRoot
  - Daily tags, available online each morning
- Users add their wagons to the next departing train
  - Activity steered by PWG conveners
- Job submission by the central framework

Trains' status

- 12% of the entire Grid activity
  - vs 6% for all individual users' jobs
- ~700 trains / month
- ~8 wagons / train
- ~2/3 run on AOD, 1/3 on ESD
- 13h turnaround time

Summary

- ALICE federates all SEs and all CPUs
  - No dedicated roles (but for the tapes)
- Network topology-aware data distribution
- Jobs go to the data
  - Remote copies as a fallback
- Complex data formats
  - Deep object hierarchy
  - Large event sizes
- Two main directions considered for improvement
  - Flattening object structure to reduce the deserialization time
  - Increasing the CPU/event ratio