ALICE Status Report

Predrag Buncic

Status Report

New T1 for ALICE

Korea Institute of Science and Technology Information - KISTI

- National research institute for information technology since 1962
  - Around 600 people working for National Information Service (development & analysis), Supercomputing and Networking
- Running High-Performance Computing Facility
  - Total 3,398 Nodes (30,592 CPUs, 300 TFlops at peak), 1,667 TB storage (introduced from 2008)

Global Science experiment Data hub Center - GSDC

- National project to promote research experiment providing computing and storage resources: HEP and other fields of research
- Running Data-Intensive Computing Facility
  - 30 Staffs: system administration, experiment support, external-relations, administration and students
  - Total 4,232 CPUs, 3,300 TB disk and 1,040 TB tape storage (accumulated since 2007)

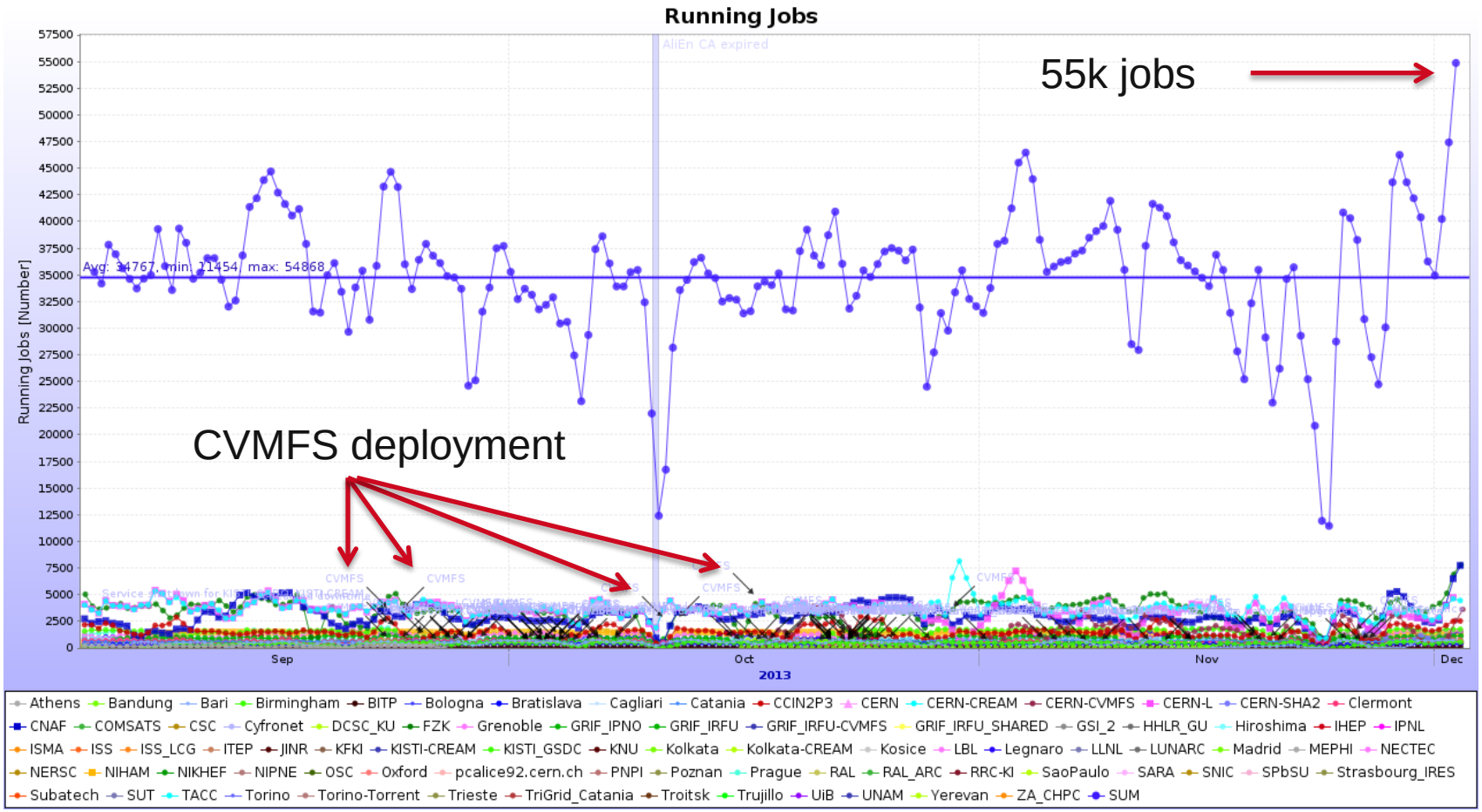
History of GSDC

- 5 years of the experience running grid computing centre with the collaboration with the ALICE

2007	2009	2010	2011	2012	2013
Formation of GSDC T2 Test-bed	T2 operation start	T1 Test-bed	KIAF	T1 candidate	

THANK YOU
감사합니다

Grid performance



CVMFS deployment

Jan 2013
……....
……
Jun 2013
……
July 2013
……
Aug 2013
……
……
……
Dec 2013
~~Apr 2014~~

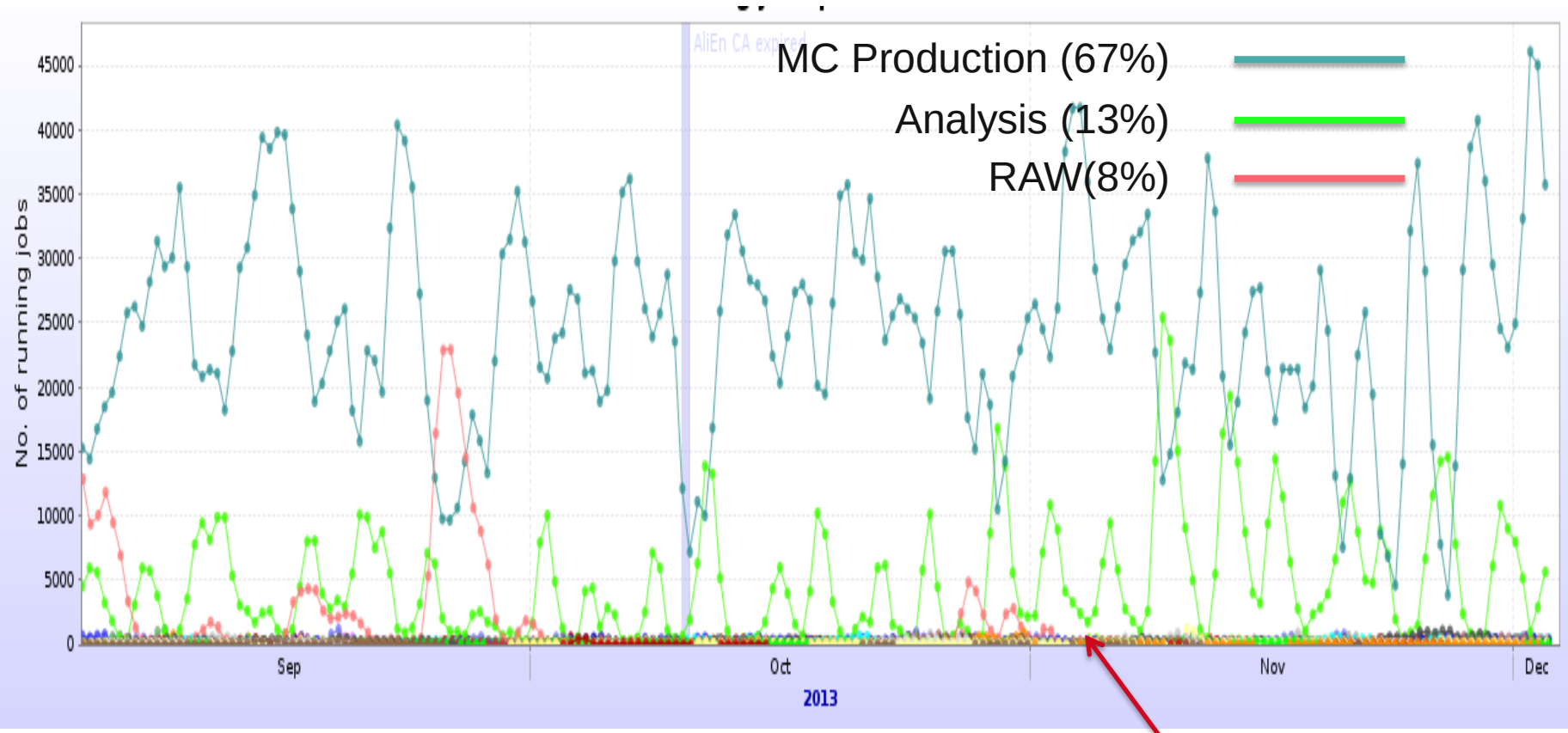
- ✓Setup Stratum 0 for ALICE
- ✓Deploy ALICE S/W on CVMFS
- ✓Migrate ALICE Repository to Stratum 1s
- ✓Test, test, test
- ✓Start deployment process on all ALICE sites
  - ✓Deploy CVMFS repository but do not use it in production
- ✓Run AliEn from CVMFS on selected site(s)
  - ✓Validate and evaluate stability, performance

Run AliEn from CVMFS on all site(s)

Deployment has to be accelerated on non WLCG sites !

61 : 9

Job Profiles



End user analysis (12%)

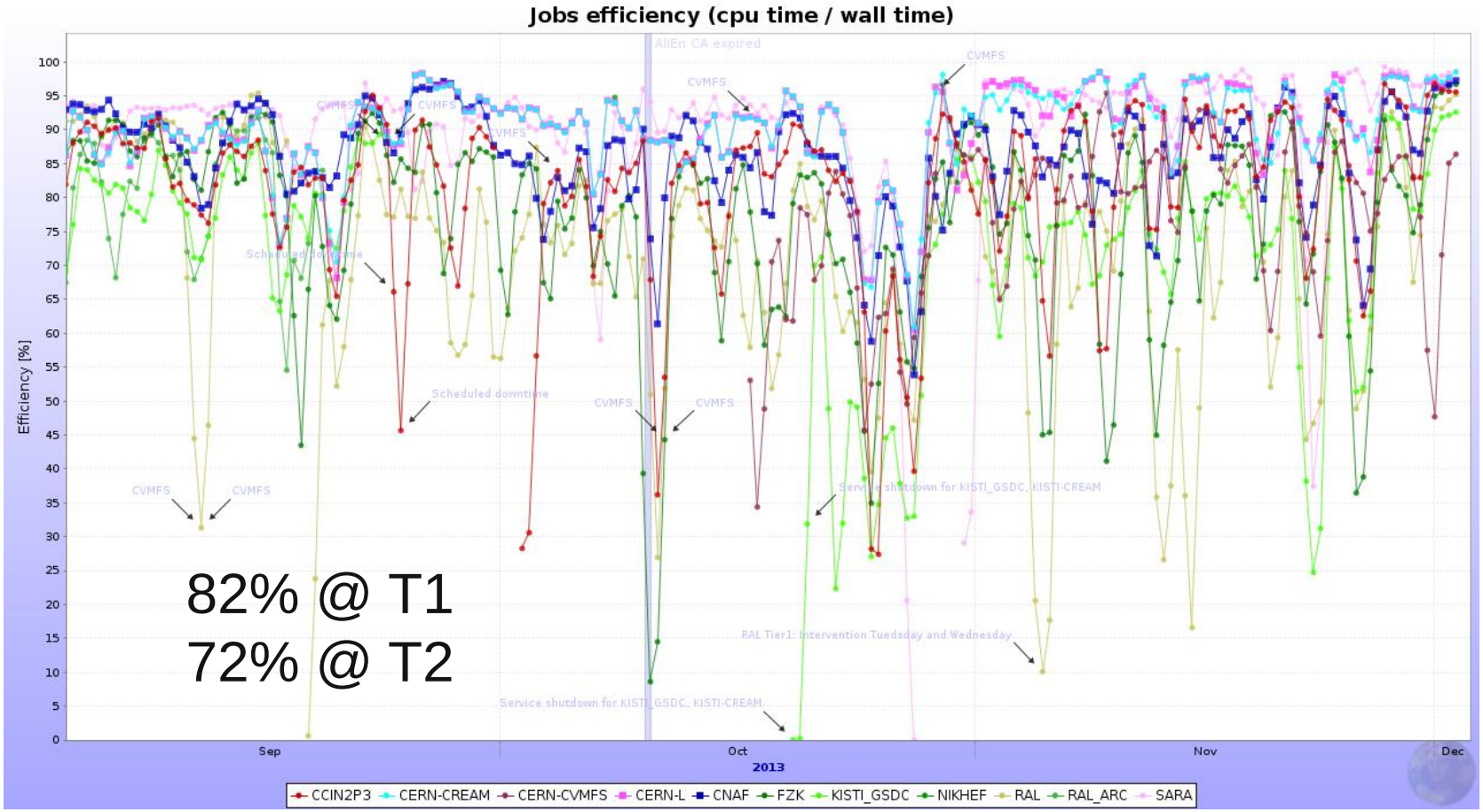
MC Production

- 113 production cycles from beginning of the calendar year
  - For comparison – 123 cycles in 2012; 639,597,409 events
  - Until end December – equal number of cycles projected
- 927,590,669 events
  - All types – p+p, p+A, A+p, A+A
  - Anchored to all data-taking years – from 2010 to 2013
  - Special emphasis on MC for data taken in 2012/2013
- No backlog on production requests

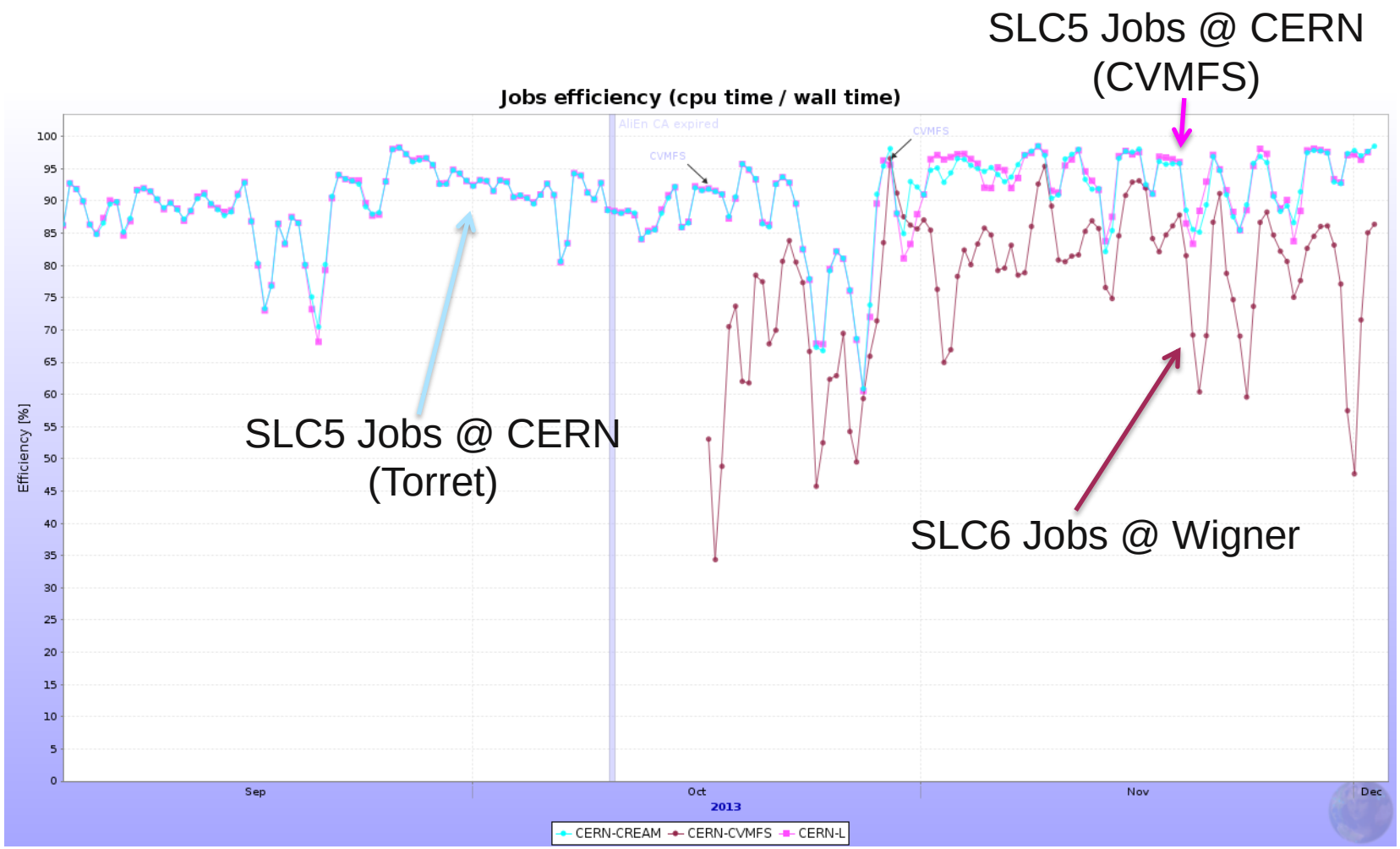
RAW data

- Completed
  - All RAW data collected in 2012-2013 is processed at least in Pass1 (some periods have Pass2)
- Plans
  - Re-processing of LHC11 p-p data (all periods)
    - With Cpass0/Cpass1
    - On standby, waiting for code updates
    - Delayed for >4 months
  - Re-processing of 2010 p-p data (selected periods)
    - To follow-up the 2011 periods reconstruction
    - Perhaps also with Cpass0/Cpass1

Job efficiency (T1s)

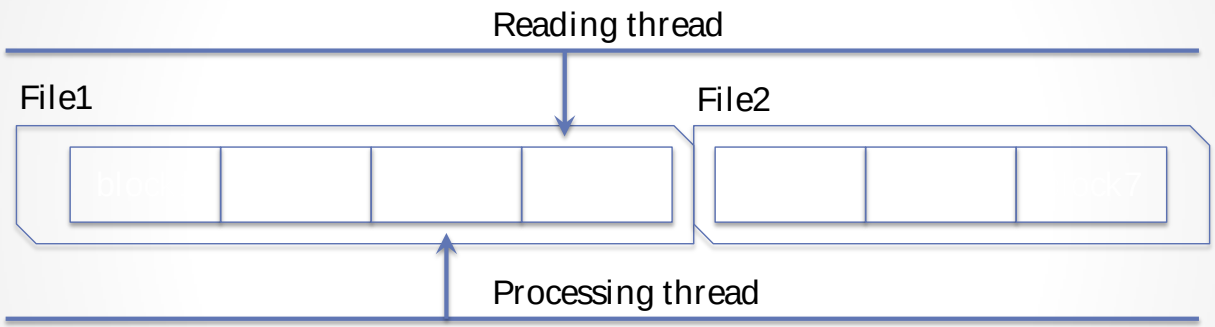


Job efficiency at “CERN”



TFilePrefetch

- Separate thread reading ahead data blocks



- First testing round found a bug
  - The fix made it to ROOT v5-34-11 (not yet used in production)
  - To be tested with LEGO trains soon
- AliAnalysisManager::SetAsyncReading()
  - Job reading from CNAF with 70% efficiency becomes 82% !
  - Forward file opening not implemented -> non-smooth transitions between files

7

Room for up to 10% improvement in efficiency at the expense of memory/disk

Summary

- Production in 2013 – OK on all levels
- Sufficient resources to fulfill all requests
- Some backlog exists on the level of old RAW data reconstruction and associated MC
- CVMFS deployment deadline advanced to end of 2014

Computing Model Update

ALICE Computing Model Update

- Scope
  - Covers period until 2017 (Run2)
  - TDR for Run3 and beyond in preparation
    - Due September 2014
- Physics goals
  - Reach the integrated luminosity of $1nb^{-1}$ of Pb-Pb collisions
  - Extend the reach of several measurements crucial for the understanding of the basic properties
  - Consolidate preliminary observations from the Run 1 data

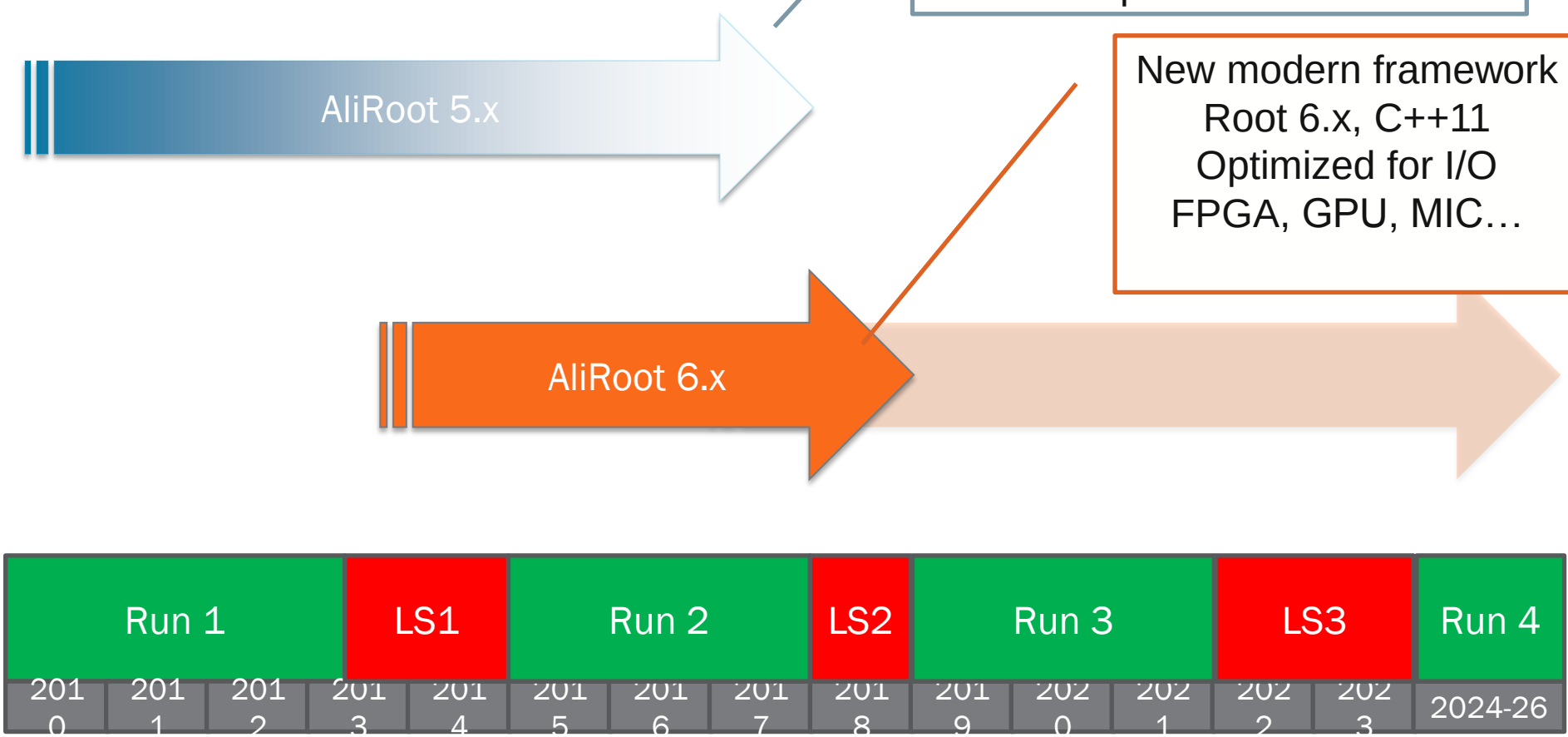
Detector Upgrade during LS1

- Increased TRD coverage (60 -> 100%)
- New electromagnetic calorimeter (DCAL)
- Upgrade of readout electronics TPC, TRD
  - readout speed x 2
- DAQ and HLT upgrade to handle increased data rates
  - 2x bandwidth to Castor (8GB/s)
- 25% event size increase
  - New detectors, higher energy

Software development timeline

Evolution of current framework
Root 5.x
Improve the algorithms and procedures

New modern framework
Root 6.x, C++11
Optimized for I/O
FPGA, GPU, MIC…



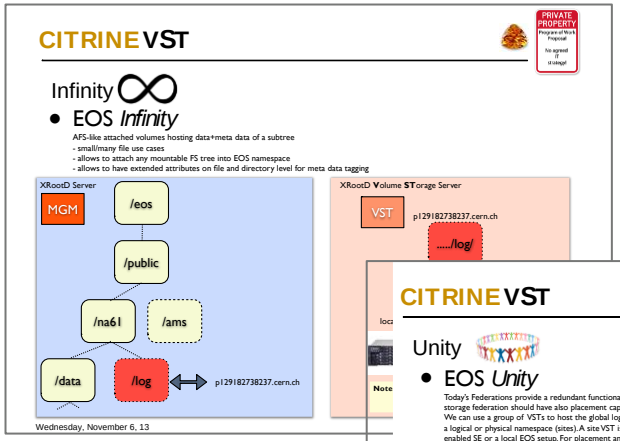
AliRoot 5.x

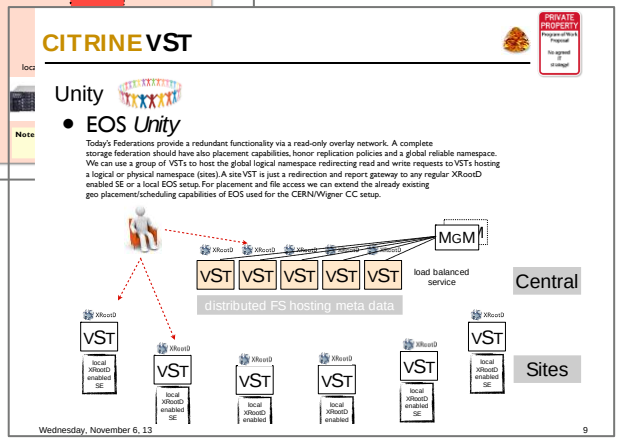
- Calibration
  - Moving time consuming Cpass0 to Online
- Reconstruction
  - Using HLT tracks seeds in Offline reconstruction
- Simulation
  - Move from G3 to G4
  - Developing and using alternative simulation strategies (fast and parameterized MC)

Distributed Computing

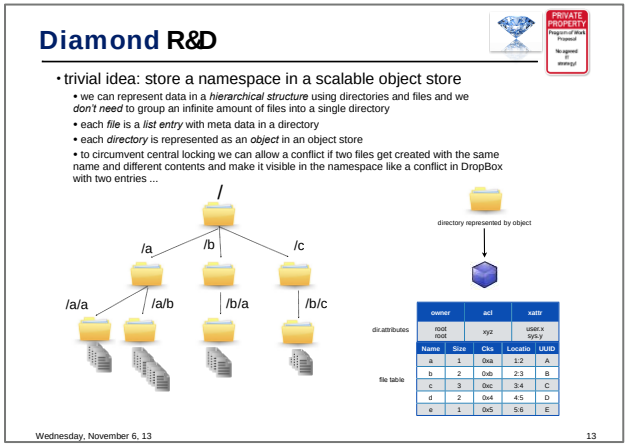
- Adapting to Cloud environments
- Use of HLT resources for Offline computing
  - ALICE HLT is still relatively small (O(1000) cores) due to extensive use of GPUs in HLT
- Improving performance of analysis trains
  - Simplifying AOD model
  - Active data management based on data popularity
- Use of HPC resources for simulation
- Long term solution for data management

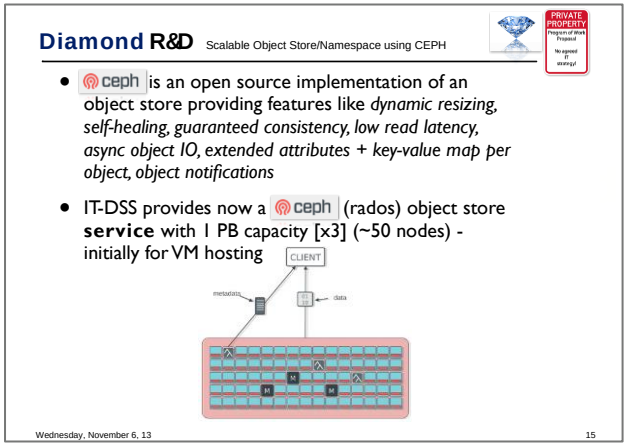
Data Management Evolution





Data Federation





We would like to explore possibility to generalize EOS architecture and reuse its components in order to create a global data federation

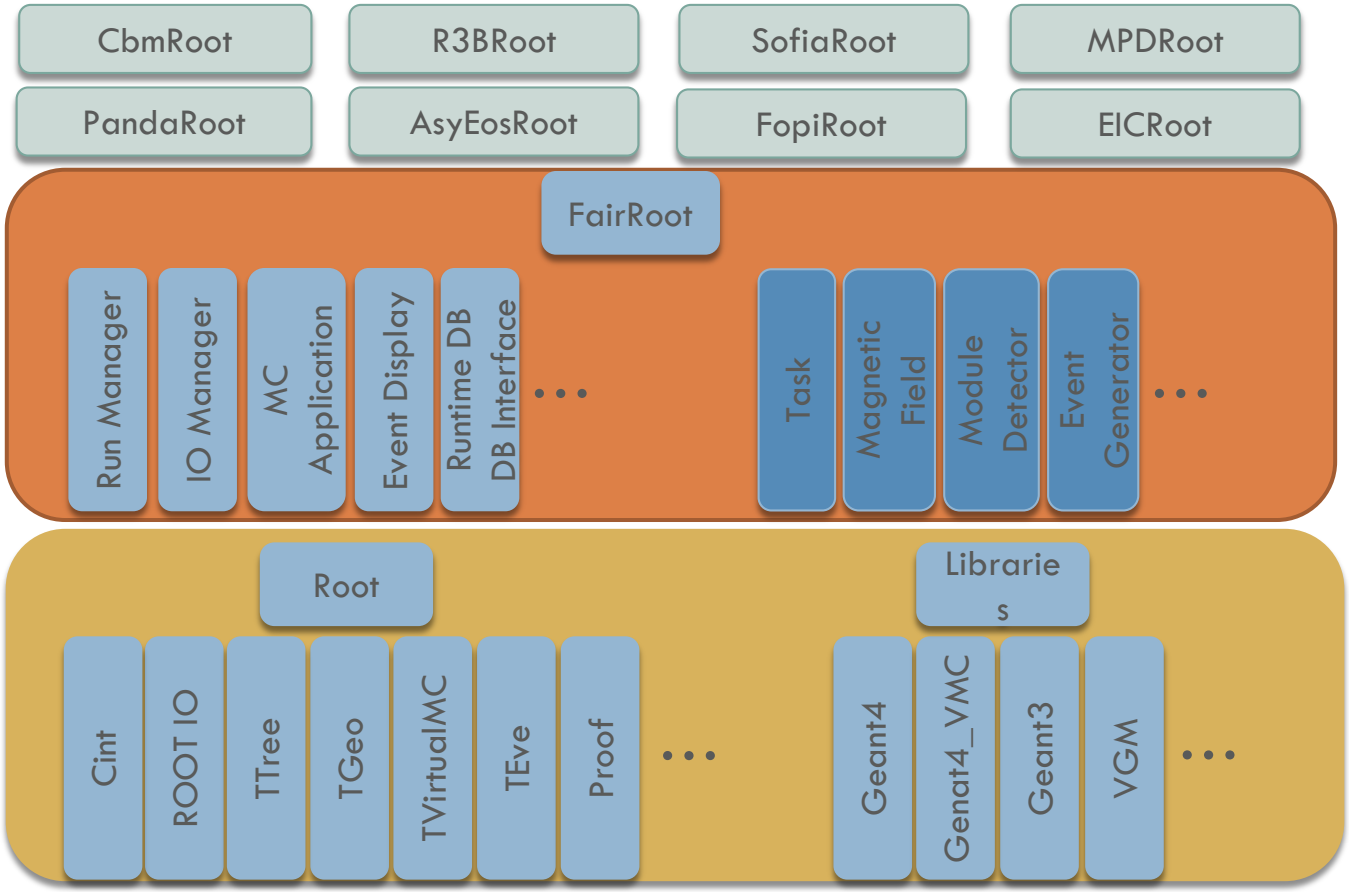
AliRoot 6.x

- New modern framework targeting Run3
- Should work in Online and Offline environment
- Based on new technologies
  - Root 6.x, C++11
- Optimized for I/O
  - New data model
- Capable of utilizing hardware accelerators
  - FPGA, GPU, MIC…
- Support for concurrency

Multi-processing vs. Multi-threading

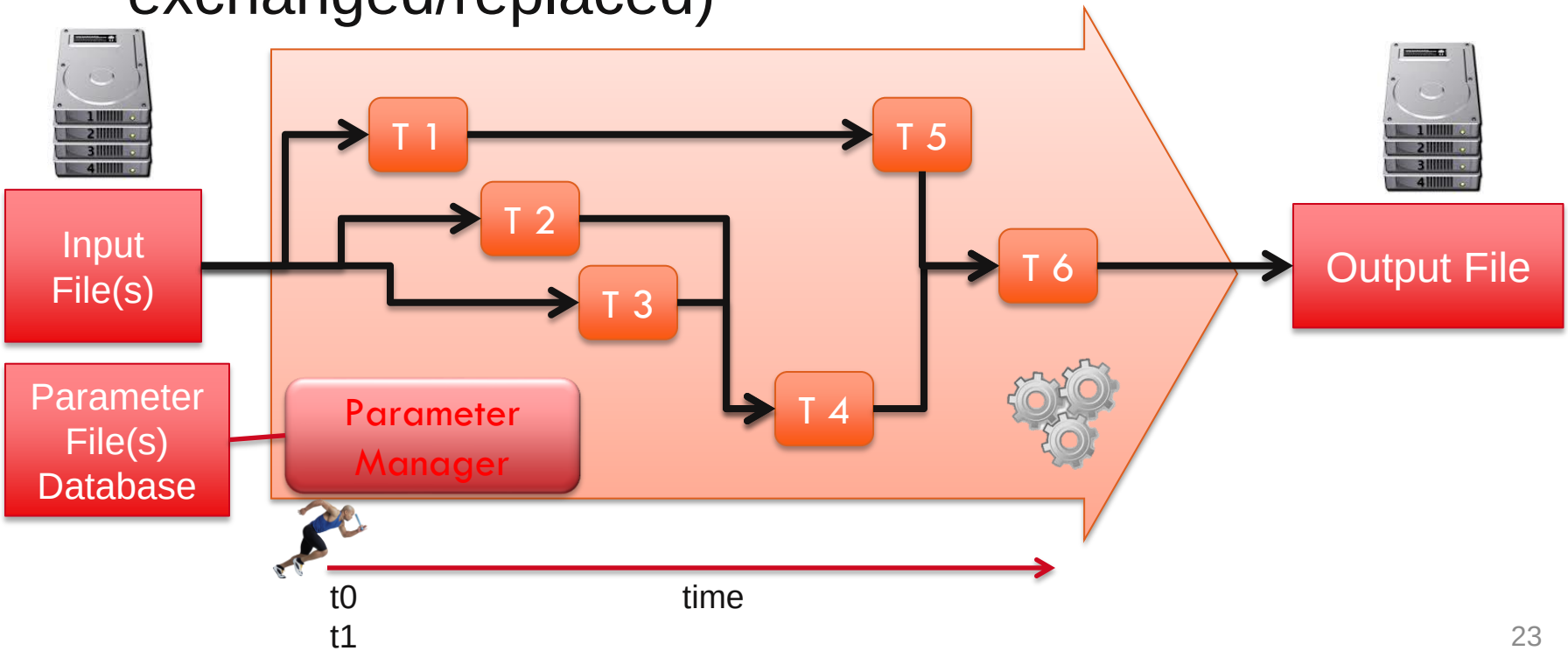
- Different processes are insulated from each other by the OS, an error in one process cannot bring down another process.
- Error in one thread can bring down all the threads in the process.
- Inter-process communication can be used across network
- Inter-thread communication is fast

FairRoot Framework

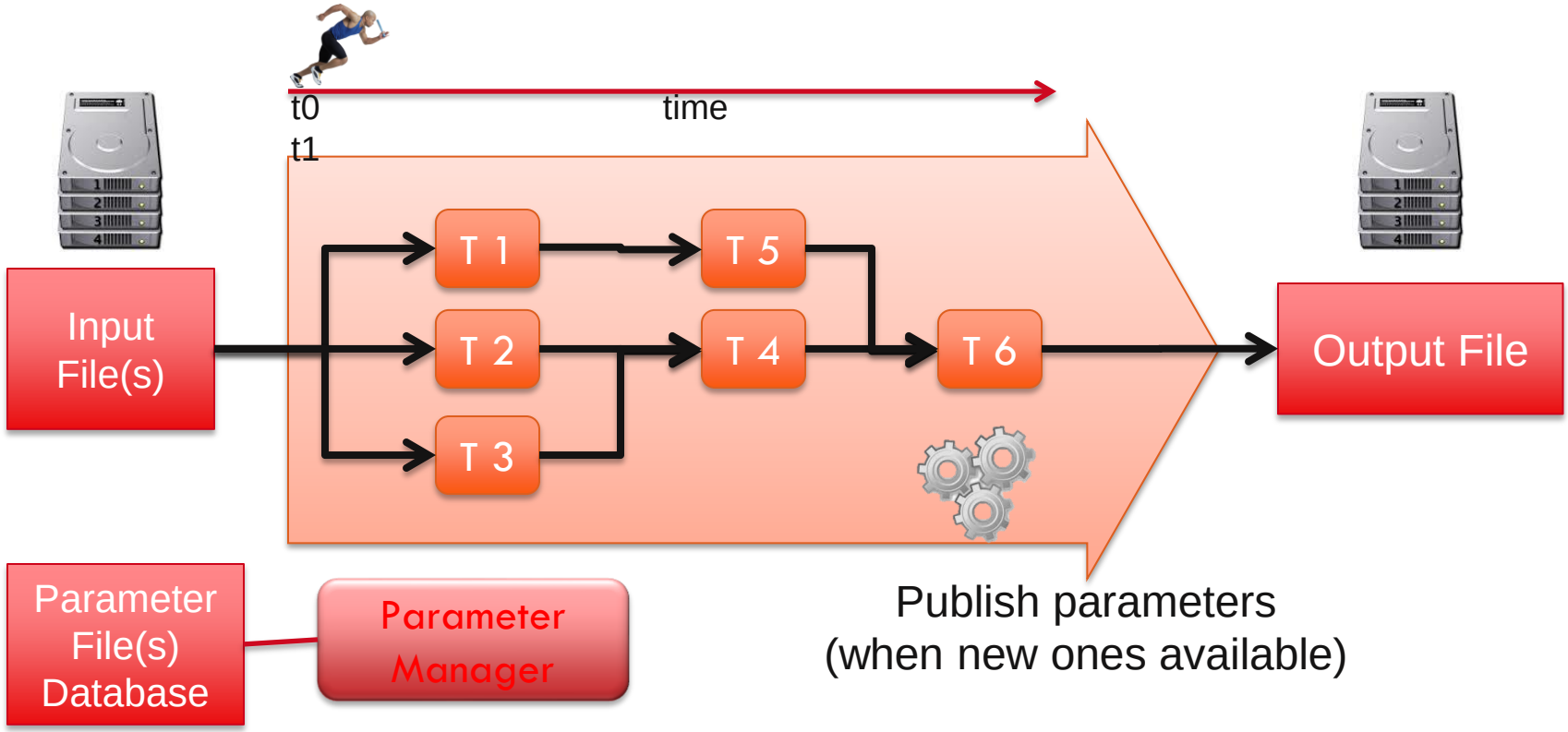


- Common software framework used by several experiments at GSI/FAIR
- Similar time constraints and data taking requirement
  - continuous readout, time frames instead of events

- ROOT event loop
- User code in Task hierarchy
- Task hierarchy runs sequentially in one process
- Tasks implement only algorithms (can be exchanged/replaced)



- Each Task is a process (can be Multi-threaded)
- Message Queues for data exchange
- Support multi-core and multi node



Next step

- Learn from others and work together
  - 0MQ approach of FairRoot looks as an appropriate solution for our Online/Offline problem
  - Possible collaboration with between ALICE and FAIR on the future framework
    - We will start prototyping new framework based on FairRoot but aiming to create a new iteration that could be used by ALICE and FAIR experiments
  - Keep an eye on all approaches to concurrency