Update of the Computing Models of the WLCG and the LHC Experiments

September 2013

Version 1.7; 16/09/13

Editorial Board

Ian Bird[a)], Predrag Buncic[a),1)], Federico Carminati[a)], Marco Cattaneo[a),4)], Peter Clarke[b),4)], Ian Fisk[c),3)], John Harvey[a)], Borut Kersevan[d),2)], Pere Mato[a)], Richard Mount[e),2)], Bernd Panzer-Steindel[a)]

a) CERN, Geneva, Switzerland
b) University of Edinburgh, UK
c) FNAL, Batavia, Ill, USA
d) University of Ljubljana, Slovenia
e) SLAC, Stanford, California, USA

1) ALICE
2) ATLAS
3) CMS
4) LHCb

Acknowledgements

Some other people ...

Executive Summary

Table of Contents

1 Introduction

Goals of this document

Goal of the activity – optimise the amount of physics output for a given level of funding: translates into event throughput per $

Need to minimise the operational costs: reduce complexity, reduce differences between experiments, reduce needs for special software (that needs support), make maximal use of provided services (eg EGI, OSG)

Use of opportunistic resources. Short periods of time (like few weeks) or overnight. Implies more or less zero configuration of compute sites is desirable.

Move to Computing as a Service – minimal config, minimal operational effort

Data preservation – where does it fit?

2 The Experiment Computing Models

2.1 ALICE

The ALICE Collaboration has recently submitted an Upgrade Letter of Intent (LHCC-2012-012) concerning the period after LS2. The computing aspects of the upgrade will be addressed in detail by a forthcoming Computing Technical Design Report due in October 2014 hence in this update we explicitly do not discuss computing after LS2 and concentrate on current practices as well as the changes foreseen for Run2 (until 2017).

2.1.1 ALICE Data model

The Figure 1 shows the data formats and the data reduction steps involved in ALICE data processing.

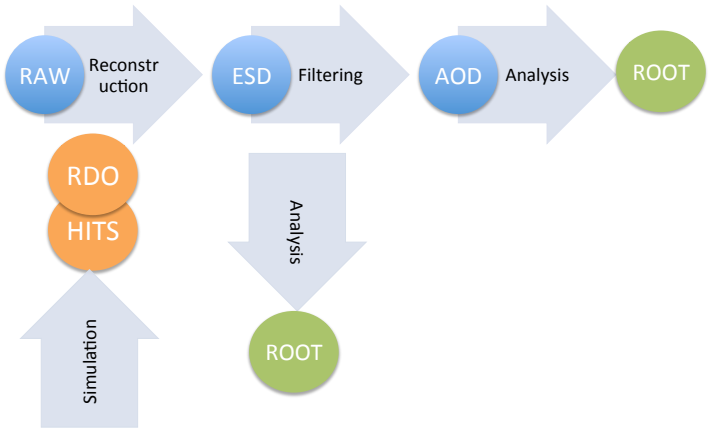


Figure 1: ALICE data types and formats

The data types are summarised in Table 1; while Table 2 shows the typical event sizes at each stage of processing and the number of copies of the data typically required for both real data and for simulated data.

Table 1: Summary of data types used by ALICE

RAW	Raw data
ESD	Event Summary Data. All information after reconstruction, sufficient for any physics analysis, AOD re-stripping and re-calibration for subsequent processing cycles. ESDs are kept with different replication factor depending on demand for the data and reconstruction cycle number.
AOD	Analysis Object Data. All events and all information needed to perform the majority of physics analysis. AODs are kept with different

	replication factor, depending on demand and version number.
HITS	Simulated energy deposits in sensitive detectors. Transient container, which is removed once the reconstruction of the MC data is completed on the worker node.
RDO	Simulated RAW data. Transient container, which is removed once the reconstruction of the MC data is completed on the worker node.
HIST	End user files containing ROOT trees and histograms.

Table 2: Event sizes and averaged replication/versioning for data and MC

Data Occupying T0/T1/T2 Storage – Planned Model (Data)					
	Event Size kB	Disk replicas of each version		Number of versions, Typical	Tape Copy
		Min allowed	Typical		
RAW	840 (p-p) 6000 (Pb-Pb) 1300 (p-Pb)	n/a	n/a	1	2x (one at T0, 1 distributed at all T1s
ESD	15 to 30% of RAW, depending on type of collision system and luminosity	1	2	One per production cycle	n/a
AOD	10 to 15% of RAW, depending on type of collision system and luminosity	1	2.6	Between 1 and 6 per production cycle	n/a

Data Occupying T0/T1/T2 Storage – Planned Model (Simulation)						
	Event Size kB	Sim Events /Data Events	Disk replicas of each version		Number of versions, Typical	Tape Copy
			Min allowed	Typical		
ESD	300 (p-p) 16900 (Pb-Pb) 1380 (p-Pb)	10% for large data sets to 30% for small data sets	1	2	1	n/a

AOD	30% of ESD		1	2.6	2	n/a

2.1.2 Data flows

The scheduled processing of raw data consists of offline calibration and update of conditions data, followed by the reconstruction and creation of ESD, AOD and Quality Assurance (QA) objects. Although the processing strategies vary between p-p and heavy-ion runs, they have in common that the first reconstruction pass must be fast and must start immediately after (for heavy-ion) or during (for p-p) data taking to allow for rapid discoveries and to quickly establish the overall properties of the collisions and data quality.

The MC and RAW data processing returns two analysis-oriented objects: Event Summary Data (ESDs) and Analysis Object Data (AODs). The ESDs contain all the information necessary for any analysis, subsequent calibration and QA checks, while the AODs contain data suitable for the majority of the analysis tasks. The size of the ESDs is 15-30% of the corresponding raw data while the AODs are about 30% of corresponding ESD size. Physics analysis can be performed from both objects, with natural preference given to the AODs. Some very specific types of analyses are performed on ESDs. Updated versions of the AODs are extracted through a skimming procedure from the ESDs, following calibration and software updates or to enrich the information available on AODs

The data collected during p-p runs, being less demanding in terms of computing resources needed for their reconstruction than heavy ion data, is processed online or quasi online during data taking at the CERN Tier 0. The data processing is preceded by a calibration cycle and is then reconstructed. The resulting ESDs and first version of AODs are kept at the Tier 0 and a copy distributed to the Tier 1s and Tier 2s. Subsequent reconstruction passes are performed at the Tier 0 and Tier 1s, after the full replication of the RAW data.

Owing to the much larger event size and complexity in heavy-ion mode, online or quasi-online reconstruction of the entire set of data would require currently unaffordable computing resources. The reconstruction of the HI data starts during the data taking, with continuous sampling of runs spread over the data-taking period. The goal of the sampling is to provide offline-quality reconstruction and QA for immediate feedback on the detector performance and data taking quality. The full reconstruction pass in the HI case is done after the data-taking end and continues up to 4 months afterwards. It is crucial that this first reconstruction ends well before the next heavy-ion run starts in order to allow for enough time to analyse the data and elaborate the new running conditions.

2.1.3 Distributed data processing

The Workload Management and Data Management systems in ALICE are based on AliEn, a set of middleware tools and services developed by the Collaboration and used for massive MonteCarlo event production since the end of 2001 and for user data analysis since 2005.

The AliEn job management services compose a three-layer lightweight system that leverages the deployed resources of the underlying WLCG infrastructures and services, including the local variations, such as EGI, NDGF and OSG.

The three layers include the AliEn Central Services that manage the whole system and distribute the workload; the AliEn Site Services that manage the interfacing to local resources and Grid services running from a VO-Box, and the JobAgents that run in Worker Nodes to download and execute the actual payload from the central Task Queue.

In a complete analogy with central job Task Queue, a central file Transfer Queue contains the list of files to be moved between storage elements and the transfers are handled by the File Transfer Daemons (FTDs) using xrootd protocol.

ALICE has used AliEn for the distributed production of Monte Carlo data, reconstruction and analysis at over 80 sites. So far more than 280M jobs were successfully run worldwide from the AliEn Task Queue (TQ), resulting in the currently active volume of 14 PB of data. All data is managed by xrootd servers and accessed via the xrootd protocol.

On the basis of the experience gained during first years of data taking and analysis and owing to the rapidly improving network infrastructure across all Grid sites, the initial hierarchical grid model has been replaced by a more 'symmetric', cloud like model. In this model, the only distinctive features of the Tier 1s are service levels and the commitment to store the data safely, commonly on mass storage systems. Tier 2s are regional computing centres with disk-only storage. University computing centres (commonly referred to as T3s) perform the same function as T2s and are fully integrated in the ALICE Grid.

The true potential of the emerging Cloud paradigm is in its inherent simplification of middleware layers that are required to obtain a Grid like view of heterogeneous and distributed resources. Readily available open source cloud middleware products are nowadays competing to solve the tasks that previously required building of a complex and expensive to maintain Grid software stack. ALICE is committed to fully exploit these emerging technologies in order to seamlessly extend its distributed computing environment to use the cloud resources where available ALICE plans to build on the results of R&D projects such as CernVM with its family of tools including Virtual Machine, File System and Cloud Federation framework. In particular, this is how ALICE plans to combine its HLT resources with CERN's private cloud infrastructure in the upcoming Run2.

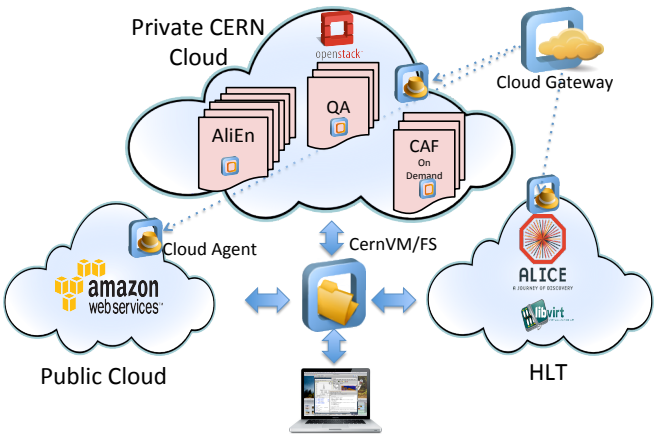


Figure 2: Using CernVM family of tools as a foundation for future ALICE grid on the Clouds

2.1.4 Non-event data

The conditions data for offline MC, RAW data processing and analysis are kept in versioned ROOT files, annotated in the AliEn catalogue. The conditions data are kept on standard Grid storage and accessed as normal files. This approach has worked well and we will continue to use it.

The AliEn File Catalog currently accounts for over a billion logical files and over 700M physical files in a MySQL database. Its current size is over 300GB. We have invested a substantial effort in the maintenance of this database that is at the heart of all our computing operations. While it still performs well under the load of 50k concurrent jobs, we are concerned about its maintenance and scalability after Run2. Thus we would like to cooperate with the other LHC experiments in identifying an existing or developing a solution that could fulfill our common needs with the necessary performance and with long-term support.

At present ALICE software is distributed to the working nodes using an ALICE developed and supported P2P protocol (BitTorrent). While again this solution worked reasonably well, we envision a transition to CVMFS in order to be in line with other LHC experiments and reduce the support needs on our side.

2.2 ATLAS

2.2.1 ATLAS Data model

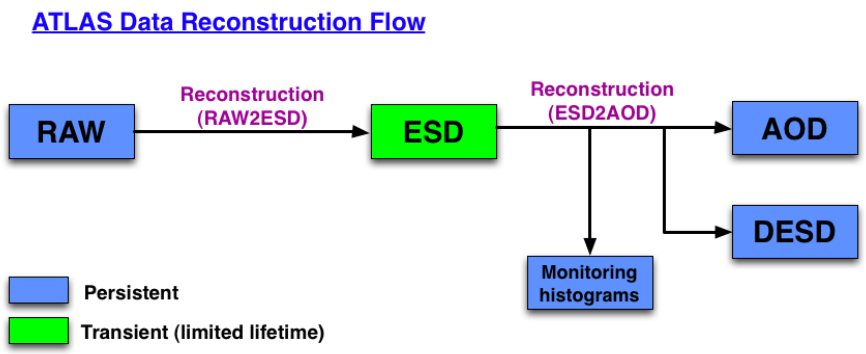


Figure 3: ATLAS Reconstruction Data Flow

The ATLAS prompt data reconstruction is performed at the Tier-0. The RAW data are processed in two steps within one job, producing first the ESDs and then the derived AODs and DESDs in the second step. The RAW data and the reconstruction outputs are exported to the ATLAS Grid storage according to the replication policy described later in the document. There is also a small set of 'monitoring data' in specialized file formats used (ROOT files) which are produced in data (re)processing for specialized studies (e.g. data quality assessment). The data re-processing from RAW is performed at Tier-1s using the same workflow.

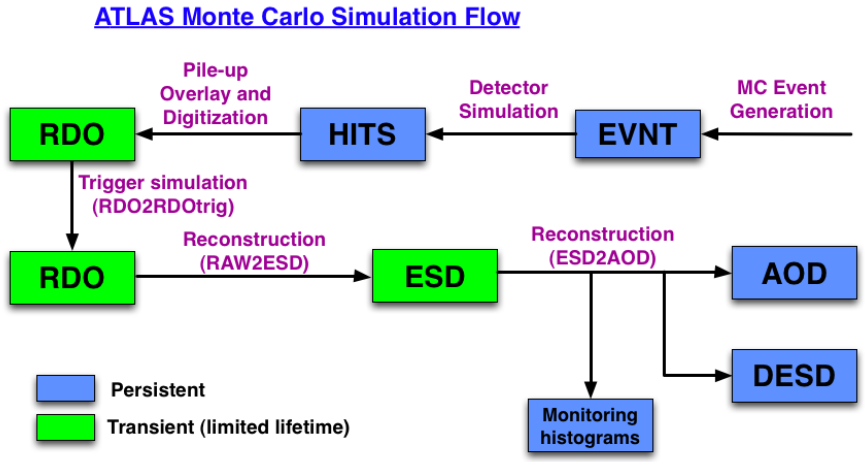


Figure 4: ATLAS Monte Carlo Simulation Data flow

The ATLAS production of the Monte Carlo (i.e. simulated) samples starts with the generation of hard-process collisions using the Monte Carlo generator programs (Sherpa, Alpgen, Powheg, AcerMC…), producing EVNT files. In some cases pre-generated inputs for the processes are produced off-Grid and registered on Grid storage for MC event generation and EVNT production. The EVNT files are then processed in the detector simulation step, producing HITS files. The modelling of pile-up is added in the next processing stage and the detector response (digitization) is simulated at the same time, producing RDO files. As a separate step, the trigger response simulation is performed again producing RDO files with the simulated trigger information added. The rest of the reconstruction chain is the same as the prompt data reconstruction. The pile-up, digitization, trigger simulation and reconstruction are usually performed together in one Grid job. The replication policy for the Monte Carlo formats is detailed later in the document.

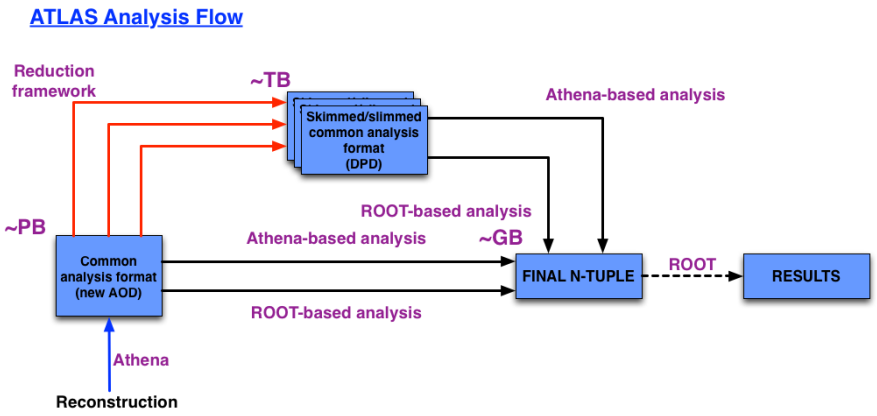


Figure 5: ATLAS Analysis Data Flow

The ATLAS Run-2 analysis starts from AODs (with an exception of specific analyses using dedicated DESDs as the input format). The data analysis can be performed within the ATLAS Athena framework or using ROOT directly to produce the final ROOT N-tuples used in individual analyses. In Run-2 the ATLAS AOD format will be updated to ensure simple ROOT readability of high-level analysis objects and thus simplify the analysis process. One can perform the analysis on the AOD files or use the reduction framework on the Grid to derive dedicated data samples in a common analysis format using skimming (removing events), slimming (removing certain types of information or collections of data objects) and thinning (removing specific objects). Details of the ATLAS analysis model and data placement are presented later in the document.

Table 3: Summary of ATLAS Data Formats

RAW	*Raw data:* a persistent presentation of the event data produced at the ATLAS online cluster (involving High Level Trigger) in byte-stream format.
ESD	*Event Summary Data:* a C++ object representation contains the detailed output of the detector reconstruction and is produced from the RAW data. It contains sufficient information to allow particle identification, track re-fitting, jet calibration etc. thus allowing for the rapid tuning of reconstruction algorithms and calibrations. Transient for data, i.e. in a rotating buffer as described in the text.
AOD	*Analysis Object Data:* C++ object representation, contains a summary of the reconstructed event, and contains sufficient information for common analyses.
DPD	*Derived Physics Data:* Skims, Slims and Thins of AOD, prepared in Group and User Analysis and specific to one or a few analysis groups; written out in ROOT N-tuple format.
NTUP	*Data in ROOT N-tuples:* In ATLAS synonymous to the DPD.
EVNT	*Event Data:* C++ object representation, contains the truth event information as produced by Monte Carlo generators (Alpgen, Sherpa, MC@NLO etc…).
HITS	*Hits data from full or fast simulation*: C++ object representation contains simulated energy deposits in active detector volumes and related particle information.
RDO	*Raw Data Object*: a C++ object representation of the byte-stream information (i.e. RAW data format), used predominantly in simulation.
DESD	*Derived Event Summary Data:* C++ object representation the DESD are derived from the ESD. Data reduction is used to select targeted events and store only the necessary information, taking into account the performance goals and rate estimates.
D(*format*)	*Further derived formats (DRAW,DAOD..):* C++ object representation of data, where reduction is used to select targeted events and store only the necessary information.

The event sizes stated in the tables below are the target event sizes assuming the LHC running conditions with beam energy of 14 TeV with a bunch spacing of 25 ns and an average pile-up rate of μ=40. As detailed later in the document, ATLAS envisages one full data re-processing from RAW and a corresponding Monte Carlo re-processing per year as well as two AOD2AOD re-processings of both real and Monte Carlo data, which brings the total number of possible versions of reconstructed object formats (AODs,DESDs) to three. The disk space usage will be optimized by using dynamic data replication to increase the availability of

popular data sets and a dynamic cleaning policy for less accessed data sets. The 'typical' number of replicas gives the number of pre-placed data replicas, including the scaling fraction of the total number of events for which the data format was produced. The 'Min. allowed' number of replicas reflects the minimal number of replicas of data sets that is allowed by the computing model and can thus go down to zero replicas of a certain version of a data set, depending on its relevance for physics analysis, as discussed later.

Table 4: Data occupation of Tier storage

Data Occupying T0/T1/T2 Storage – Planned Model (Data)					
	Event Size kB	Disk replicas of each version		Number of versions, Typical	Tape Copy
		Min allowed	Typical		
RAW	1000	1	1	1	2
ESD	2700	0.2	0.2	1	0
AOD	350	0	2	3	1
DPD	varies	0	1	3	0

Data Occupying T0/T1/T2 Storage – Planned Model (Simulation)						
	Event Size kB	Sim Events /Data Events	Disk replicas of each version		Number of versions, Typical	Tape Copy
			Min allowed	Typical		
EVNT	100	5	0	2	1	1
HITS	1000	1.4	0	0	1	1
RDO	3700	1.4	0.05	0.05	1	0
ESD	3700	1.4	0.05	0.05	1	0
AOD	550	1.4	0	2	3	1
DPD	varies	1.4	0	1	3	0

2.2.2 Anticipated event rates and data streams

In 2012 the total average trigger rate during proton-proton collisions in stable beam conditions was ~550 Hz. This includes all physics streams (prompt +delayed), which, with an average compressed RAW event size of 0.8 MB/event, gave an average 440 MB/s streaming rate of RAW events.

For 2015-2018 ATLAS uses as the baseline an average trigger rate of 1 kHz as event input to the Tier-0 for prompt reconstruction. For 25 ns bunch spacing, the average event size will be similar to the Run-1 (2012) values, at least in the first year when the pile-up average value at 25 ns is unlikely to exceed μ~25 and

would go up to about 1 MB/event at the average pile-up of $\mu \sim 40$. The anticipated average streaming throughput of RAW data in Run-2 is thus between 800 MB/s and 1000 MB/s.

The primary motivation for the doubling of the HLT output rate is the desire to keep the effective single electron and muon p_T threshold at 25-30 GeV and therefore well below the Jacobian peak from W boson decays. This strategy was very successfully employed for Run-1 and these triggers were used by the majority of the ATLAS analyses. The single lepton triggers allows ATLAS to keep an inclusive and unbiased trigger for most electro-weak processes, for example associated production of the Higgs boson in the WH channel. Options for relying on more exclusive selections are under study, but generally incur efficiency losses for many physics channels and carry an increased risk of missing new physics signals. At mu=40, the rate of the single lepton triggers alone are expected to reach 4-500 Hz with the rate dominated by irreducible W and Z boson decays. This can be compared to a typical average in 2012 of approximately 120 Hz. Other triggers will also see increased rates, but this increase is expected to be more modest as thresholds will need to be raised to keep the L1 rate below the maximum possible.

In Run-1 (2012) ATLAS had 3 main physics streams: 'egamma', 'muon' and 'JetTauEtmiss' that accounted for almost all the prompt physics triggers and three delayed streams: 'b-physics', 'HadDelayed' which is mostly lower thresholds jet/etmiss triggers and 'JetCalibDelayed' which is a small stream used mostly for jet calibration purposes. The overlaps between the prompt streams was typically 10% ('b-physics' delayed stream has a stronger overlap with 'muon' prompt stream and 'HadDelayed' stream with 'JetTauEtmiss' stream by design). For Run-2, ATLAS is at present investigating whether to keep the same prompt stream structure and what gains could be achieved by changing it. At present it is not foreseen to record delayed streams in Run-2.

2.2.3 Role of the Tiers

In Run-1, the Tier-0 facility at CERN was utilized for prompt data reconstruction along with the supporting activities like the prompt data calibration, data quality determination etc. The Tier-0 was in Run-1 also used as the primary archival (tape) repository of ATLAS RAW data as well as the promptly reconstructed data in AOD and DESD format. Promptly reconstructed data in ESD format were archived on tape only for some small data streams dedicated to activities like data quality assessment, calibration and alignment processing, or detector performance studies. Substantial bandwidth and processing effort at Tier-0 during Run-1 were also dedicated to the compression of RAW data, which could not be done online, but by which the volume of persistent RAW data and its replicas on disk could be almost halved.

The role of the Tier-0 facility is expected to remain unchanged in Run-2. RAW compression at Tier-0 will not be necessary anymore, this is foreseen to be incorporated into the online DAQ workflows.

Preparations are on-going for the scenario of over-spilling part of the prompt data reconstruction to the Tier-1 centres in case periods of severe resource congestion at the Tier-0 would appear in Run-2. For that purpose it is necessary

to synchronize the task and job definitions, configurations and creation mechanisms used at Tier-0 and in the Grid production system. The requirements on the alignment between the Grid and Tier-0 production systems are based on the experience from Run-1, when the over-spilling was tried for the first time for the processing of Heavy Ion collisions and then certain difficulties were identified. In order to ensure the readiness of Tier-1s for the prompt data processing a small stream of data will be constantly processed at Tier-1s and validated by comparing the outputs with the Tier-0 processing outputs.

In Run-1 ATLAS used part of the available resources at CERN as the CAF (CERN facility for data calibration and alignment). The CAF function included providing servers, non-automated data calibration, partial reconstruction, debugging, monitoring and other expert activities as well as high priority group and user analysis activities within the Grid environment. During Run-1 ATLAS developed procedures to simply and dynamically allocate the idle resources in Tier-0 to the CAF Grid environment (e.g. during LHC technical stops) and to reassign CAF resources to the Tier-0 in case of processing congestions. This functionality is expected to remain also in Run-2. In addition, ATLAS commissioned the High Level Trigger (HLT) farm to work in the ATLAS Grid environment for simulation production, using the Open Stack IAAS interface. In Run-2, ATLAS is also planning to use the HLT farm, when idle, in the Grid environment. The HLT farm is expected to be reconfigured within an hour. ATLAS expects to use the fraction of the HLT farm not taken by the ATLAS Trigger/DAQ activities for Monte Carlo production. Considering that a typical ATLAS simulation job is aimed to take 12 hours, the reasonable minimal length of available time for it to be worth re-configuring would be one day; these scenaria will be explored further.

Developments in the ATLAS software and the distributed computing environment, as well as the technological progress manifest in the ATLAS Grid sites, have enabled ATLAS to perform workflows formerly restricted to Tier-1s (data re-processing, group analysis, Monte-Carlo reconstruction) also at most of the Tier-2s at the end of Run-1. This advantage is expected to be expanded upon in Run-2, enabling ATLAS to perform all the required workflows across the ATLAS Tier-1s and Tier-2s and thus to optimize the load and throughput on the sites by providing each an optimal mixture of jobs in terms of high and low throughput, memory consumption and CPU load. The Tier-1s will still have the distinguished role as the data archival centres, providing the tape archival service for the second copy of the RAW data exported from CERN as well as the archival of the ATLAS data deemed crucial for data preservation. In addition, the Tier-1s will remain the centres for the most memory and I/O intensive computing tasks.

In Run-1, the ATLAS Tier-2s were used to perform the bulk of Monte-Carlo event simulation and user analysis. Based on operational experience, the Tier-2s were also categorized into several sub-categories using the connectivity and reliability metrics automatically collected by the ATLAS distributed computing monitoring. This categorization enabled ATLAS to optimize the data placement with respect to its criticality and use-cases. This optimization is expected to remain in place and be expanded further in Run-2.

With the technological progress of wide area networks and consequently improved network connectivity ATLAS has already in Run-1 gradually moved away from the hierarchical association of Tier-2s to one Tier-1 (MONARC model) and is now able to associate workflows between a well-connected Tier-2 and several Tier-1s, based on monitored connectivity metrics. This functionality will be further pursued in Run-2 and is being incorporated into the production system and distributed data management developments.

In Run-1 ATLAS has benefitted from the availability of substantial beyond-pledge and opportunistic CPU resources. These additional resources proved extremely valuable, allowing ATLAS to pursue an even richer and more precise set of physics results than would otherwise have been possible in the same time frame. Our resource planning is based upon the physics programme that can be accomplished within achievable pledged resources, corresponding to a 'flat' spending budget, while we hope that our centres and funding agencies will continue to provide ATLAS with the invaluable resources beyond those pledged that will allow us to accomplish an optimal research programme and physics productivity.

In 2012 and now during Long Shutdown 1, ATLAS Computing has also further explored and is investigating the use of opportunistic resources, either 'Cloud'/IAAS resources or High Performance Computing (HPC) centres and the creation of a transparent and straightforward interface between these and the ATLAS distributed computing environment. The current development work is implementing the connection either directly through the PanDA infrastructure or, in case of HPCs, also through the ARC middleware interface to PanDA. In 2012 and 2013, ATLAS Computing has already demonstrated successful use of the Amazon EC2, Google Computing Engine 'Cloud/IAAS' and Helix Nebula resources for Monte-Carlo event simulation.
In Run-2 ATLAS will continue to exploit and enhance this versatility of using opportunistic resources to complement our existing resources to off-load CPU intensive and low I/O workloads which under-use our I/O optimized grid resources. A typical example would be 'Monte Carlo generation of hard-process events' (Alpgen, Sherpa Monte Carlo event generators), which took ~15% of CPU resources on the ATLAS Grid sites in 2012. Another target use is the CPU intensive Geant4 simulation. The Grid resources can then be better used for I/O intensive work: (simulation, reconstruction, group production, analysis), important especially during peak demand.
In addition, if ATLAS Grid sites want to move to use cloud middleware (e.g. OpenStack) or provide dual-use resources ('batch'+HPC) the ATLAS versatility gained by the ongoing R&D provides this option.

Based on the Run1 experience, ATLAS is upgrading and re-modelling both its distributed workload and data management systems.

2.2.4 Distributed Computing Environment

The new data management system (Rucio) will introduce more granular handling of ATLAS data (file replicas rather than dataset replicas) together with a more intelligent implementation of data ownership for users and groups. This will allow optimization of the space usage while offering more functionalities in terms of local and global quotas, thus e.g. relaxing the need for dedicated group

storage (SRM space tokens) at sites. The modular structure of the new system will allow better interfacing with newly developed middleware clients and services such as FTS3 and gfal2, allowing the utilization of different protocols than SRM for file transfers/access/deletion and storage management (e.g. HTTP for LAN and WAN access, WebDAV for storage management, xRootd for WAN and LAN access, plain gridFTP for data transfers). Additionally, ATLAS has been commissioning a xRootd based federated storage infrastructure, which would offer more resilience in data access (increasing job efficiency) and allow transparent file access through the WAN, reducing the need of massive data replication and therefore optimizing disk space usage. A HTTP based federation model, while still not in a mature enough state, is also being evaluated for the long term, because of the benefits in utilizing a standard protocol such as HTTP rather than an HEP-specific one.

The new workload management system will still rely on the pilot-based PanDA service as core component. It will be complemented with a higher modular layer offering the functionalities of job definition and management (JEDI) and workflow and task management (DEfT). Briefly, JEDI will allow dynamic job definition for better matching of site capabilities with payload requirements. It will also allow a better handling of existing use cases such as file merging, reducing the requirements for network resources and storage I/O bandwidth. DEfT will allow users and production managers to more flexibly define high level workflows and improve the automation and reliability of the system, speeding up the job processing sequence and minimizing the time needed to complete production and analysis tasks. The combination of JEDI and DEfT (prodsys2) will soon replace the existing ATLAS production system and will offer a unique platform for distributed production and analysis, reducing the cost for maintenance and support, while offering new functionalities to end users.

For the longer term, ATLAS is investigating the possibility to further refine the granularity of the production and analysis payload from a collection of events to a single event. A prototype of such an “Event Server”, combining functionalities of Panda/Prodsys2, the xROOTd federation and AthenaMP, is under development. Such an innovative system would allow to enable and/or optimize the usage of (mostly) opportunistic resources where job pre-emption cannot be avoided, such as cloud resources, opportunistic HPC resources and even user desktops through volunteering computing. The workflows where the gain of this approach in processing efficiency is highest are high in CPU and low in I/O, such as Geant4 simulation.

The increased processing times and memory footprints of the ATLAS software due to higher energy and pile-up in Run-2 processing suggest a different approach to efficiently utilize the resources provided by the WLCG. A simple model with one job executing on one computing core is not sufficient any more due to increased memory requirements. AthenaMP provides a solution to efficiently run multi-processing jobs, using less memory, which will be exploited further by the future concurrent framework (foreseen to be ready during Run-2). The current commissioning scenario of multi-core jobs assumes each site to allocate a pool of resources to exclusively run multi-core jobs. The optimal number of cores to be used is at present evaluated to be eight but further studies will be performed to understand the optimal scenario. This will be extended by a

dynamic setup, where the batch system will allocate multi-core resources on demand to be able to balance between single and multi-core jobs automatically. Most of the batch systems used by WLCG sites already support dynamic resource allocations and some sites already use it in production in 2013. The full deployment of multi-core setup on all sites will follow the schedule of AthenaMP deployment with the commissioning of the ATLAS Run-2 software environment.

2.2.5 Data Categories

With the introduction of the automated data deletion algorithms and dataset lifetimes within the ATLAS distributed data management in 2010, the data replica types were categorized into two main categories:

- Primary: the base replicas guaranteed to be available on disk. These replicas are not subject to automatic clean-up.
- Secondary: extra replicas dynamically created according to the usage metrics. The replicas are subject to the automatic cleanup in case of disk space shortage at the site and are thus considered as volatile. Within a site, the dataset replicas least accessed by Grid analysis jobs are the first candidates for deletion.

ATLAS distributed computing can thus control the minimal (i.e. non-volatile) number of appropriately placed data replicas and supplement the demand for more replicas by using the dynamic data replication (PD2P) mechanism and the automatic deletion agent of the secondary replicas (Victor), as determined by operational criteria and physics analysis needs. The same mechanisms are expected to remain in Run-2, with further improvements and refinements.

2.2.6 Replication Policies

ATLAS in Run-1 controlled the static replication of the pre-placed data replicas with the policies established by the ATLAS Computing Resources Management. The replication policy was periodically reviewed at least on a yearly basis based on gained operational experience, disk and tape space constraints and ATLAS computing model evolution. A continuing process is envisaged also for Run-2. The static replication policies involve the RAW data, real data ESDs, AODs and DESDs and simulated EVNT, HITS, RDOs, ESDs, AODs and DESDs.

As in Run-1, for Run-2 ATLAS foresees several yearly re-processings of data. For real data this will be either full re-processings from RAW to AOD or fast AOD2AOD re-processings, which are a new feature for Run-2 and will be described in more detail below. For simulated data, there will be one or more consecutive Monte-Carlo productions within one year of data taking, either starting from scratch, i.e. producing new EVNT files and then repeating the full simulation chain up to AODs, or, for full Geant 4 simulation, starting from the archived HITS files and repeating the digitization and reconstruction chain.

As already stated, RAW data collected in Run-1 is kept as two tape copies, one in the CERN tape archive and the other one distributed across the ATLAS Tier-1s. This will remain unchanged during Run-2. In addition, a single copy of RAW data for the current data-taking year will be retained on disk at Tier-1s. This proved to be very useful for trigger and other studies involving sparse event picking procedures in Run-1. In addition, with the scenario of over-spilling the prompt

reconstruction to Tier-1s the presence of RAW on disk becomes essential for preventing tape buffer congestions and efficient prompt processing at Tier-1s. At the end of 2012 ATLAS reduced the volume of Run-1 RAW data on disk to 5% of the total volume, and will keep it at this fraction during Long Shutdown 1. In Run-2, a full single copy of the current year's RAW data will be kept on Tier-1 disk again, as was done in the Run-1 data taking years.

In Run-1, ATLAS introduced a primary replica of the real ESD data with a lifetime of 4-6 weeks in the Tier-1s, primarily for the use of expert reconstruction developers, to allow for the central production of derived data formats (DESDs) for physics analysis and data quality assessment. The volume corresponded to approximately 20% of the current year's data. The success of using a rotating disk-resident buffer of real ESD in Tier-1s in Run-1 gives confidence that the same procedure can be used in Run-2. During the Long Shutdown 1 the ESD volume of Run-1 data on Tier-1 disks has been reduced to 5% of the total volume, consisting of a representative subset of Run-1 data and small special streams for trigger studies and detector performance studies, calibration and alignment processing, data quality assessment, and similar activities.

The simulated EVNT files will initially be placed in two primary replicas in Tier-2s and distributed dynamically as needed across the Tier-1s and Tier-2s to provide the optimal input distribution for the ATLAS Monte-Carlo simulation. In addition, a custodial tape copy will be made at Tier-1s.

The simulated HITS from full Geant4 simulation, being very CPU intensive to make, will be stored in one custodial tape copy on Tier-1s. The HITS from other (faster) flavours of ATLAS Monte-Carlo simulation will not be stored and the fast simulation production chain will write out only AODs.

The simulated ESD and RDO (where RDO is the simulation equivalent of RAW), will be in Run-2 kept as a persistent copy in Tier-2s only upon explicit request for specific simulated samples. They have proven to be essential to the ATLAS Combined Performance and Trigger groups in Run-1. Based on the Run-1 experience, the total simulated ESD volume kept in a persistent copy is estimated to correspond to 10% of the total simulation statistics and the RDO fraction is assumed to remain constant at 5% of the total statistics.

In the updated model foreseen for Run-2, the number of pre-placed AOD replicas kept in Tier-1 and Tier-2 disks for both real and simulated data will be reduced as much as possible, placing more reliance on the dynamic data replication. At the end of Run-1 the number of pre-placed (primary) AOD copies was reduced to two copies in Tier-1s and two copies in Tier-2 disk, as well as one custodial tape copy at Tier-1s.

In Run-2, The AODs and DESDs from the most recent (real data) (re-)processing and the corresponding simulation AODs will be pre-placed in two primary copies, one at Tier-1s and one at Tier-2s, as well as a custodial tape copy at the Tier-1s. During the year, the number of primary AOD and DESD copies will be dynamically reduced down to zero (no primary disk copies) as described in detail later in the document.

There is also a small set of specialized file formats (ROOT N-tuples) which are produced in data (re-)processing for specialized studies (e.g. data quality

assessment), which are also exported to the Tier-1s as primary copies. Their disk volume is small (negligible) and their replication policy and lifetime is handled by the needs of the ATLAS Data Preparation Coordination.

The log files corresponding to any distributed computing workflow are merged into files big enough for tape storage and stored to tape in one custodial replica.

The analysis derivation process may involve skimming (removing events), slimming (removing certain types of information or collections of data objects) and thinning (removing specific objects). The derived group analysis DPDs (ROOT N-tuples) are stored in the disk space allocated to the ATLAS physics, combined performance, trigger, … groups and are managed individually by the groups.

In Run-2, the groups will keep their dedicated disk space same as in Run-1 but the group disk space needs are expected to decrease proportionally to the real and simulated AOD volume with the software and operational improvements of Group analysis and the AOD format evolution performed during the Long Shutdown 1.

In Run-2, as in Run-1, the final user analysis files, produced by individual users, are expected not to be stored on pledged ATLAS resources. Temporary storage (scratch) space will be provided for the users to retrieve the files produced on the ATLAS Grid resources.

With the introduction of dynamic data replication, producing secondary copies of datasets, performance metrics for data placement have been put in place to guarantee good data accessibility. These metrics steer the "PanDA Dynamic Data Placement" ("PD2P"), the automated, popularity-based replication mechanism for both AODs and derived formats (group analysis DPDs). In Run-1 this approach has demonstrated its ability to optimize group/user analysis throughput using a low number of pre-placed replicas, improved CPU usage and decreased the overall network bandwidth and disk space demands. The same mechanism will therefore remain in use in the Run-2 period.

The ATLAS data distribution plan consequently depends critically on allocation of adequate disk space for buffers for dynamically-placed, secondary copies of data sets at both Tier-1 and Tier-2. Based on the Run-1 experience, the required disk space represents approximately 30% of the total disk space volume.

2.2.7 Deletion Policies

The number of primary copies of AODs corresponding to different (re)processings and simulation productions in Tier-1s and Tier-2s will be decreased from two to zero according to their relevance and popularity for physics analysis. While in Run-1 this was done only via a human decision process, in Run-2 auxiliary automated mechanisms based on popularity will be introduced to provide an additional dynamic component in reducing the number of pre-placed primary AOD and DESD replicas in Tier-1 and Tier-2 disks. The policy envisaged for this automated mechanism is:

- If a data set was not accessed for six months set the Tier-1 replica as secondary,

- If a data set was not accessed for one year set the Tier-2 replica as secondary.

This algorithm will thus reduce the disk copies of unused data sets on disk to zero and only custodial tape replicas will remain. The actual time windows will be fine-tuned with the Run-2 operational experience, the six and 12 months are given as tentative starting values.

A similar algorithm will monitor the usage of the derived group analysis data (DPDs) in the group disk space allocations and provide adequate warning to the groups to optimize their disk space usage. The group space deletions will be scheduled manually by the groups owning the group space.

In addition, as a further correction algorithm, an automated procedure to re-distribute the primary copies of very popular datasets to disk resources with enough free space, sufficient CPU resources and good network accessibility will also be put in place.

As a consequence of this model update ATLAS will in Run-2 more actively use the tape resources for data retrieval than in Run-1.

It should be stressed that during Run-1, with the introduction of ATLAS-developed PD2P and automated deletion mechanisms, ATLAS was able to reduce the number of pre-placed AOD copies considerably, especially in Tier-2s, where 20 copies of AODs were used in 2010 (10 current+10 previous), 10 copies in 2011 (8 current+2 previous) to 4 (2 current+2 previous) at the end of Run-1.

Another substantial improvement in Run-2 will be the handling of 'transient' data sets, which are files in the processing workflows that are subsequently merged into bigger files, to optimize the file sizes for data transfer and storage, before being deleted. These 'transient' data sets in Run-1 occupied a substantial fraction of both the buffering disk space on the Grid and of the network transfer bandwidth. Thus, optimizing this workflow in Run-2, which will be introduced in the new ATLAS production system (PanDA with JEDI and DEfT) and the new ATLAS Distributed Data Management (Rucio) will allow more efficient use of these resources, e.g. for additional dynamically created replicas of popular data.

2.2.8 Description of Workflows

2.2.8.1 Prompt reconstruction

RAW data are exported from the ATLAS on-line computing facility to the Tier-0 resources at CERN, where the RAW data is reconstructed and ESD, AODs, DESDs and a series of dedicated ROOT N-tuples are produced. RAW and derived data (except ESD from bulk physics streams) are stored on CERN tape and the RAW, ESD and AOD are exported to the Grid sites according to the replication policy described above. The CPU resource needs for the Tier-0 are determined from the requirement on the fraction of events that must be processed during the live time of the LHC (and corresponds roughly to the maximum LHC live time that can be sustained over a period of several days, i.e. 70% in Run-1). The CPU requirement thus scales linearly with the HLT trigger output rate and the reconstruction time per event.

In Run-2 the new ATLAS Grid production system and the Tier-0 processing environment are being aligned to be able to use the same reconstruction job

configurations. In case of congestion of CERN resources this would enable ATLAS to reconstruct fractions of data also at Tier-1s in a straightforward way if needed.

2.2.8.2 Data re-processing from RAW

In Run-1 the RAW data was reprocessed at Tier-1s (and a set of Tier-2s in 2012 as a proof of concept) on average once per year with an improved reconstruction software release. At the same time the same software release was introduced in the Tier-0 data processing to ensure the consistency of the whole data set. The reprocessed AODs, DESDs and ESDs were again distributed with a similar distribution policy as the promptly reconstructed data, but in this case only the most interesting 5% of the ESD volume is stored and exported to the Grid. In Run-2, about one re-processing from RAW data per year is envisaged and it will be complemented by several AOD2AOD re-processings described below. Simulated samples are reprocessed (re-digitized and re-reconstructed) in synchronization with the data re-processing.

2.2.8.3 AOD2AOD Data re-processing

In Run-2 the state of the art reconstruction and fixes will be applied in dedicated AOD2AOD re-processings, which will happen a couple of times per year. This new model has several advantages, in particular more frequent incorporation of relevant improvements in the analysis software. Also, the series of reconstruction improvements which were in Run-1 applied at the group analysis level and thus repeated several times for each group analysis format, will in Run-2 be applied only once at the central AOD2AOD re-processing level.

The contents of the AOD that can be modified in a AOD2AOD re-processing are the high-level reconstruction objects such as jets and missing E_T. Changes to the reconstruction of lower-level objects such as tracks and clusters cannot be made using only the AOD - such operations would require a RAW to AOD re-processing.

The AOD2AOD re-processing CPU consumption per event is expected to be an order of magnitude faster than the full RAW to AOD re-processing and consequently the CPU resources are not expected to be a limiting factor, ATLAS will however need to maintain a vigilant replication and deletion policy to keep the use of disk resources within reasonable constraints. This consequently motivates the updated data replication and deletion policies as described in this document.

2.2.8.4 Monte Carlo production

The volume of simulated data required for analysis has historically been hard to predict. The excellent LHC and ATLAS performance in Run-1 resulted in the ability to address a much wider range of physics analyses, with a higher level of precision, surpassing the most optimistic expectations. In addition, detailed physics studies established that the simulation is of unprecedented quality compared to previous generations of experiments, describing the data very well in most analyses; this quality opened-up more 'safe' applications for the simulated data. These two facts together significantly enhanced ATLAS physics output in 2011 and 2012, and they motivated production of larger than foreseen

simulation statistics. Consequently, in Run-1 the ratio of required simulated to real data was of the order of two to one, and the required volume of simulated data might be even higher in Run-2 with higher required precision for background estimation and an even larger phase space for New Physics searches.

Due to its high CPU cost, the outputs of full Geant4 simulation (HITS) are stored in one custodial tape copy on Tier-1 tapes to be re-used in several Monte-Carlo re-processings, as required. As already stated, the HITS from faster simulation flavours will be only of transient nature in Run-2. More details in the developments in optimizing the Monte-Carlo simulation are stated in the software section.

The simulated HITS are processed in the digitization step to simulate the detector response. In this step the pileup is overlaid on top of the hard-scattering process, either by using dedicated simulated HITS of the 'minimum bias' pile-up processes or by overlaying real data events recorded in a dedicated data stream; the latter was already used at the end of Run-1 for almost all heavy ion (Pb-Pb collisions) analyses which study high-pT physics with photons, W/Z bosons, and jets. The CPU requirements of the digitization and pileup overlay are on the same order as simulated data reconstruction; there is an on-going work to improve the CPU consumption in the anticipated harsher pile-up conditions in Run-2. The digitization outputs (RDO) are transient and only stored upon explicit approved requests for specific performance studies and software development.

The Monte Carlo reconstruction uses the same software release and configuration as the real data reconstruction, with the addition of the trigger simulation, the latter adding 15% to the CPU time per event and to the AOD event size. In order to minimize the memory consumption, the trigger response simulation is performed as a separate step after pile-up simulation and digitization, using RDOs as input and again producing RDO files with the simulated trigger information added. The ESDs produced in simulation reconstruction are again stored only upon explicit approved requests for specific performance studies and software development.

For the very fast simulation types, the digitization and reconstruction CPU time becomes dominant. The developments during the Long Shutdown 1 are targeting the development of fast-digitization and fast-reconstruction algorithms by using e.g. truth-seeded procedures to provide a complete very fast full Monte-Carlo production chain. More compact output formats are being explored, to allow taking advantage of the high Monte Carlo simulation and reconstruction speed to produce very large Monte Carlo samples.

Simulated samples are reprocessed (re-digitized and re-reconstructed) in synchronization with the data re-processing. As in Run-1, these occasions will in Run-2 also be used to further improve the pile-up and trigger simulation. Consequently, in Run-2 the AOD2AOD re-processings introduced for the data will also be used for the Monte Carlo samples.

An unexpected development at the end of Run-1 was that a non-negligible CPU consumption was taken by the Monte-Carlo 'event generation' of hard processes with the introduction of large-scale use of the advanced (multi-leg, Next-to-Leading-Order..) Monte Carlo generators (Sherpa, Alpgen, Madgraph..) which in 2012 and 2013 represented 15% of the total ATLAS pledged CPU resources.

ATLAS is aiming to use opportunistic resources (HLT farm, HPC centers) for these high CPU and low I/O tasks whenever possible and we aim to significantly reduce the amount of standard grid CPU resources used for event generation though the use of these opportunistic resources, especially HPCs, in spite of the growing need for more sophisticated (and therefore slower) MC generator configurations in Run2.

2.2.8.5 Group-level analysis

In the Run-1 period the ATLAS Physics Analysis and Combined Performance groups organized their analysis workflow into processing the data and simulation samples from AODs into dedicated (group and analysis specific) data processing, mostly producing DPDs in ROOT N-tuple format. Initially, these 'group productions' were made by dedicated user jobs and the group data produced were stored in assigned group disk spaces on the Grid. In order to provide better control over the software used and to guarantee consistency and more effective data placement, this group production activity was centralized by introducing a new Group Production working group and related Group Production workflow on the Grid at the end of 2011.

The Group analysis workflow, i.e. the centralized production of analysis formats targeted for specific sets of analyses (e.g. top quark measurements, Higgs boson property measurements, SUSY searches…) has been a success in terms of consolidating separate group workflows and of common/coherent software usage. It also incorporates many features of the re-processing improvements, to some extent reducing the need for full data and Monte Carlo re-processing, in principle giving a net gain in Grid CPU utilization, but at the cost of an unforeseen and considerable increase in disk resource needs for storing the derived Group analysis outputs. ATLAS realized that considerable resource gains were possible by revising the analysis workflows and software and produced a detailed study of how to improve the ATLAS analysis for Run-2. The principal findings, which are being incorporated in the ATLAS software and Distributed

Computing operations are:

- Rewriting the AOD persistency format to provide ROOT-readable high level reconstruction objects, such as jets or muon tracks without the need of the ATLAS Athena framework.
- Development and maintenance of a centrally supported reduction framework, which allows the skimming, slimming, thinning of the new AOD analysis format, enabling groups to produce derivations suitable for their analyses.
- Development and maintenance of a unified ROOT-based framework and an Athena-based framework for physics and combined performance analysis with a common analysis Event Data Model, which facilitates the use of common analysis tools.

In view of these improvements in Run-2 the volume of the derived group analysis formats will become more manageable and is envisaged to occupy twice the total volume of one AOD replica. In combination with the AOD2AOD re-processing, the changes should also considerably reduce the CPU consumption and processing duplication. A detailed study has shown that space savings up to

a factor four should be achieved by the introduction of these improvements compared to this area of the Run-1 computing model.

2.2.8.6 User-level analysis

In Run-1 the ATLAS user analysis was performed either directly on the AODs (20% of 'users' used AODs) or on the derived group analysis formats. In Run-2, the 'users' will still be able to run their analysis directly on AODs (in the new ROOT-readable format) or use the improved derived skimmed, slimmed and thinned group analysis data in the same format.

ATLAS analysis has been very actively performed in the Grid environment and considerable CPU resources are allocated to this activity (in the year 2012 20% of all ATLAS CPU resources were used by user analysis).

2.2.9 Differences for Heavy Ion (or p-Pb)

Heavy Ion analyses in Run-1 used ESDs and hence derived DPDs/N-tuples as their input analysis format. The aim is to develop a dedicated format with a lower event size for Run-2. In terms of CPU and disk resources, the ATLAS Heavy ion program in total used 10% of the overall resources and it is expected to stay within these limits also in Run-2.

2.2.10 Non-event data

2.2.10.1 ATLAS Databases in Run-2

It is common in ATLAS, to think of its 'data' as primarily that which is stored in event-wise files; this data comprises, by far, the bulk of the total ATLAS data volume. But a non-trivial volume of critical data is stored in databases: needed to record events (including configuration and calibration) and then downstream to deploy, execute, and monitor event processing and understand results. These data are not only critical to the above-mentioned event recording and processing but also support many essential applications that do not necessarily process events.

There is a significant amount of software and infrastructure needed to enter and maintain the data within databases and make it available to many diverse applications as well as to develop and optimize those applications. Moreover, data stored in databases can and is used to refine the processing of events in a few important ways. A job deployed on the grid is wasting grid resources if it opens a file which contains no events of interest to the task when a simple database query can be used to make that determination. Reading many standard ATLAS data formats requires database access to get information which is not contained in event-wise files. Moving excessive volumes of in-file metadata to database systems is a mechanism to reduce file volume.

The information needed from databases for the majority of our systems fits well into relational database models or has been adapted to do so. Moreover, ATLAS has invested significant design and programming resources into building subsystem specific schemas and infrastructure to efficiently store and deliver data needed to the diverse applications that need it. In saying this, we absolutely do not exclude the use of alternative technologies when they provide functionality in areas where relational DBMS falls short of requirements, such as

in the areas of scalability or accessibility (demands for the data is high and widely network distributed). For example:

- Grid-wide access to conditions, trigger and geometry data: we deployed Frontier/Squid servers at all our Oracle offline sites to provide efficient caching of conditions data from Oracle to event-wise processes running on the grid.
- The web-based CVMFS distributed file system is used to guarantee access to all conditions data files from all sites where ATLAS software is installed (as the technology is the same).
- Access to aggregated or partitioned quantities: various structured storage alternatives were explored and successfully deployed within ATLAS Distributed Computing to provide performant and scalable access to aggregated information, generating reports and providing effectual information for monitoring. Currently, the favoured alternative storage being deployed is based on Hadoop.

While the above enhancements do not relieve our relational storage systems of its role as the master repository for the data, they relieve the relational DBMS systems of workload, leaving capacity for workflows where alternatives are not possible or not yet developed.

For the deployed systems in the relational DBMS, we have accumulated significant experience and legacy data during Run-1, which is useful to make projections of the needs for Run-2. Database volume is one aspect of the depth and complexity of the data stored in databases. The Run-1 ATLAS data volume for all schemas in the 3 main ATLAS database servers was in June 2013: 6.8 TB in ATONR (the online database), 18.7 TB in ATLR (offline database, storing a wide variety of operational systems such as the Conditions database) and 15.3 ADCR (offline database, storing distributed computing related schemas for file and job management).

In addition, there is a database instance ATLARC at CERN, which serves as the database for TAG files (event-level information) in Run-1, using 16.2 TB. In addition to the CERN instance, two further instances were in Run-1 available at the ATLAS Tier-1s. In Run-2, the plan is to streamline and optimize this database, potentially using new technologies like Hadoop, thus rendering this particular instance obsolete.

In Run-2, we do not expect the online ATONR DB to grow much because the larger schemas have sliding windows removing older data (which is always fully replicated on ATLR). For ATLR, we expect monotonic growth at roughly the rate seen in Run-1 (so volumes may grow by a factor of 2 to 3 for Run-2). ADCR is much more difficult to predict, given the uncertainty in the offline event rate and size, number of files, as well as the current major evolutions of the DQ2/Rucio and ProdSys-II systems. The LFC service and DB will not exist any longer but Rucio will include its functionality.

In Run-2, ORACLE support will not be needed at all T1s. Only a few T1s will continue to provide ORACLE and Frontier services. The reduction in number of required sites is a consequence of on improved overall stability of the distributed system.

2.3 CMS

2.3.1 CMS Data model

The CMS data model is show schematically in Figure 6, showing the various data formats and processing steps between them.

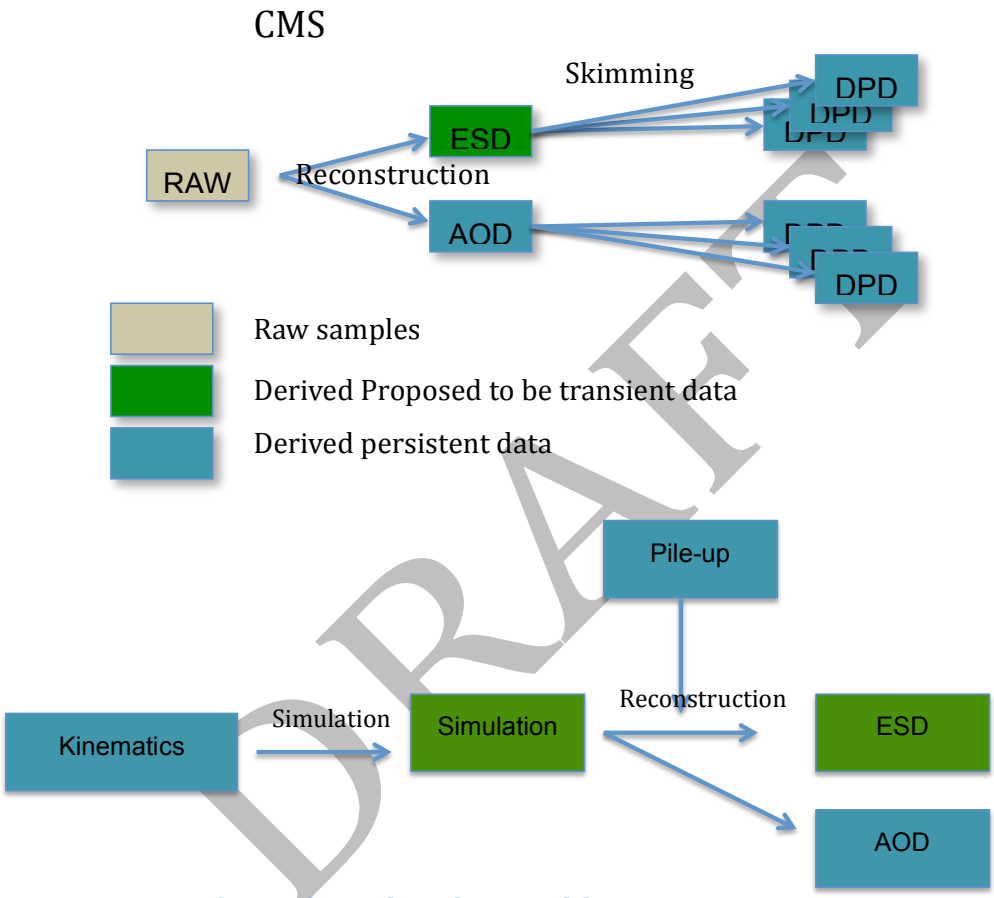


Figure 6: The CMS data and Simulation Models

The CMS prompt reconstruction and reprocessing passes look similar. An application with multiple output modules is executed where the first step is reconstruction and is followed by a skimming step to select data sets specific to different analyses. The ESD (RECO data) is proposed to be transient for most datasets in 2015. The primary use of the larger ESD format is detector commissioning, which can be done with a transient disk-only copy with a limited life expectancy of 6 months to a year. There are a limited number of analysis applications that have a physics reason to need access to ESD formats, which will be stored persistently. The various data formats are listed in Table 5.

Table 5: CMS Data formats

RAW	Raw data
ESD	Event Summary Data. All information after reconstruction, including information content of RAW. Now considered not cost effective to keep systematically. (Also referred to in CMS as RECO data)
AOD	Analysis Object Data. All events. Enough information to re-run many reconstruction algorithms.
D3PD	Derived Physics Data (3rd generation). Skims, Slims and Thins of AOD, likely specific to one or a few analysis groups. (Also referred to in CMS under the umbrella USER defined tier)
NTUP	Ntuples
HEPMC	Generator Output
HITS	Simulated energy deposits in sensitive detectors
RDO	Simulated RAW data (Also referred to in CMS as GEN-SIM-RAW)

Table 6 explains the event sizes and typical numbers of replicas for real and simulated data for CMS.

Table 6: Data sizes and number of replicas (real and MC data)

Data Occupying T0/T1/T2 Storage – Planned Model (Data)					
	Event Size kB	Disk replicas of each version		Number of versions, Typical	Tape Copy
		Min allowed	Typical		
RAW	500	0	1	1	2
ESD	1000	1	1.1	2.5	1
AOD	300	2	4	2.5	1
D3PD	Varies	2	4	2.5	1
NTUP	Varies	1	2	2.5	0

Data Occupying T0/T1/T2 Storage – Planned Model (Simulation)						
	Event Size kB	Sim Events /Data Events	Disk replicas of each version		Number of versions, Typical	Tape Copy
			Min allowed	Typical		
HEPMC	1000	1.3	0.1	0.1	1	1
HITS	Temporary					

RDO	1500	1.3	0.1	0.1	0.1	1
ESD	1100	1.3	0.1	0.1	2	1
AOD	350	1.3	2	4	2	1
D3PD	Rarely Produced					
NTUP	Varies	1.3	1	1	2	0

On average CMS data events are processed 2.5 times during a year. The first is a real time prompt-reconstruction, and before final publications there is a second full reconstruction with new calibrations and sometimes a new software release. During the year there are targeted reprocessing passes for individual datasets that don't add up for more than 0.5 passes. One of the accomplishments of 2012 was to arrive at a point where only 2.5 complete processing passes were needed. We expect that in 2015 we will maintain discipline similar to 2012 to contain the reprocessing needs and fully utilize the Higher Level Trigger farm during the shutdown to augment the existing resources. During the first year of Run1 many more processing passes were needed.

The simulation, also described in Figure 6, is similar to data processing, but it has a few additional elements. The kinematic step is either performed at run time directly or performed in advance with a dedicated program and written in specialized LHE files. These events are then simulated to get the detector response. The underlying physics events are combined with in time pile-up from the same collision and out of time pile-up from a window of previous and following collisions. For Run2 this step can involve combining 1000 minimum bias pile-up events and requires enormous IO to maintain high CPU efficiency. To mitigate this CMS is preparing complete crossings in advance that will be reused with difference physics events. The complete event is reconstructed and AOD and ESD formats can be written.

2.3.2 CMS Workflows

The general CMS workflows are described below

- The data is transferred from P5 as a temporary binary format and reformatted into RAW Root files. This process is called repacking in CMS. The compression in Root reduces the file size by about 20%. The RAW files are moved to EOS space and to CASTOR for archiving.
- A small portion of the data, typically 40Hz, is identified as the Express data, which is repacked and reconstructed immediately in small jobs. This data is made available to users within an hour of data collection. For Run2 this data will be available worldwide through the use of the data federation. In Run2 these samples were only accessible through the CAF at CERN.
- A Tier-0 prompt calibration loop accesses the Express data and performs some automated calibration. Hot channels in the silicon and the beam spot position can be calculated continuously. Several calibrations are

calculated during a 48-hour hold loop before reconstruction jobs are launched.

- Prompt Reconstruction jobs are launched on the RAW data files 48 hours after the data is collected. The reconstruction jobs produce both RECO and AOD formats and register the samples for subscription and access by users. In 2015 these jobs will be launched centrally, but will be executed at the Tier-0 and the Tier-1s in roughly equal numbers to deal with the higher trigger rate of more complex events.
- Skimming workflows are launched on the RECO and AOD files at Tier-1s to produce dedicated analysis datasets. Skims are encouraged to select fewer than 10% of the initial sample; larger selections use the full primary dataset.
- Simulation jobs are sent to Tier-2 centres to perform the bulk of the simulated events. Any spare resources at Tier-1 are also used for simulation.
- Tier-1 and Tier-2 centres participate in simulation reconstruction, a process that combines additional interactions to generate a crossing, and reconstructs the physics objects to produce the AOD and RECO formats for simulation.
- Users launch analysis jobs at Tier-2 centres, and in 2015 to available resources at Tier-1s.

2.3.3 CMS Workflow Improvements

CMS has invested heavily in I/O optimizations within the application to allow efficient reading of the data over the network. Over local I/O protocols the application does not need to stage the input file to the worker node disk. This work has also allowed reading the data over the wide area network using the xrootd mechanism while maintaining a high CPU efficiency.

The RAW and AOD data formats have benefitted from programs to introduce efficient compression. The RAW data has historically been smaller than the original estimates. The AOD is compressed and an evaluation is on going to assess which objects are not heavily used and could be dropped. The target for reduction is 30%. At the same time CMS is also studying adding elements to the AOD to allow user level recalculation of particle flow elements. This would increase the size of the AOD by a similar factor of 30%, but would potential eliminate some targeted centralized reprocessing passes.

2.3.4 Anticipated event rates and data streams

The increasing trigger rates are motivated by a desire to maintain similar thresholds and sensitivity to Higgs physics and to potential new physics. We estimate that with increase in instantaneous luminosity as well as the increase in cross section will raise the trigger rate in CMS to be approximately 1kHz, unless there are significant triggering improvements. This is shown in comparison to 2012 in Table 7.

Table 7: CMS trigger rate comparison 2012-2015

Year	Rate Prompt	Rate Delayed (also known in CMS as "Parked")	Output
2012	460Hz	360Hz	328MB/s
2015	1000Hz	0Hz	600MB/s

CMS has run with inclusive data streams since the beginning of Run1 and expects to continue with this model. There are 15-20 physics datasets and an overlap between them of approximately 25%. Given that 2015 is the first year of the run and our current estimate is that all available resources will be needed to process the events we collect and promptly reconstruct, we do not anticipate writing additional triggers for delayed reconstruction. These events might not be processed until 2018.

2.3.5 CMS Data flows

The main CMS data flows are the following:

- The data is selected and reformatted into RAW data files using the Tier-0 resources at CERN. A copy of the data is archived at CERN a second copy is sent to archive services away from CERN. A second copy is also transferred to disk resources at Tier-1 centres, either from CERN or from the remote archives based on PhEDEx routing decisions.
- The events in the RECO format (ESD) will be written to disk-based resources and maintained for several months for detector commissioning and specific analyses. The AOD formats will be written to disk and sent to an archive service for permanent storage.
- AOD and RECO will be served out to Tier-2 centres to provide more access and opportunities for processing. The replication and retention will be based on the popularity of the dataset.
- Simulation samples from Tier-2s are continuously transferred to the Tier-1s to disk storage. Once the final processing steps are complete and the data is validated it will be replicated to the archive.
- Group ntuples are not normally replicated by CMS provided services unless the samples have be elevated to official datasets through a validation and publication procedure.

2.3.6 Role of the Computing Tiers

CMS is expecting to break down the boundaries between the computing tiers in time for the restart in 2015. The main changes with respect to the original computing model is that the Tier-1 centres will perform a significant fraction of the prompt reconstruction and to participate in user analysis, while also continuing in the role of data and simulation reprocessing. Tier-2 sites will be involved in MC reconstruction, while continuing a central role in simulation production and user analysis. The Tier-0 facility is not expected to change in capability or expectation. Tier-1 sites in CMS will be classified according to their level of support and their operation of central services. Tier-1 sites are

expected to operate central services like CVMFS Stratum 1 services and Frontier squids, FTS, and potentially pilot factories. Tier-1s are expected to operate 24/7 and respond to issues out of normal business hours. Tier-1 centres may be collocated with Archival Services, but may not be. Archival services are tape-based systems located at a sufficient number of locations to give 2 copies of the RAW data in separate facilities. We do not want to remove existing high availability centres, but new Tier-1s may be brought up with a higher share of CPU and disk and compensated with archival resources elsewhere. Tier-2 centres will have similar capabilities to Tier-1 centres, but are not expected to run services for other sites and are expected to operate with business hour support.

A significant addition for Run2 is the role of the Higher Level Trigger farm (HLT). Beginning in 2013 CMS commissioned the HLT for use in organized reprocessing. The HLT has been demonstrated at 6000 cores running data reprocessing reading the samples over Xrootd from EOS at the CERN computing centre. The cores are made available through the OpenStack cloud interface provisioned through the Glide-In WMS pilot system. From the production perspective the HLT is simply another large site, though no traditional grid services are offered and the pilot system handles resource provisioning of virtual machines. Currently this farm in roughly a third of total Tier-1 processing capacity, and is expected to grow with farm upgrades needed for Run2. In order to fully process one year's data in Run2 the HLT will play an important role by allowing CMS to process the data for a year in 3 months during the regular winter shutdown. The HLT is only fully available for offline processing during shutdown periods, so CMS will need to be prepared when the farm is ready with new releases and new calibrations.

2.3.7 CMS Replication Policy

By the restart in 2015 CMS expects to replicate data produced automatically and release caching space based on the level of access. The goal is that if a user requests any sample produced in the current run the system will be able to provide access to that dataset and there will never be an error thrown that the sample is not available on a site the user has access to. In order to meet this goal CMS will rely on automatic replication and transfer approval to a large pool of centrally controlled disk space at Tier-2s and the full disk at Tier-1 centres, and CMS will make use of xrootd over the wide area to serve data to locations that have available CPU for users but do not have a copy of the sample requested. The goal in CMS is that the vast majority of data will continue to be served from local copies but that during the completion of transfers, or to serve unpopular samples, data can be streamed to a running application.

2.3.8 CMS Distributed Computing

The use of more centrally controlled storage space will also dictate the use of more central control of the location jobs are submitted to. CMS will still accept the use of white-list and black-lists but will treat them as advisory. Up to now CMS has allowed the user to completely determine where and how they will choose to work.

A considerable fraction of the CMS analysis access to data is used in the production of group ntuples and datasets. These samples tend to be produced by individuals but used by a larger community and are similar to the organized processing workflows. CMS is investigating two potential models to achieve more organization in large-scale analysis production. The first is to treat the group production as just another step in the regular production. The second is to develop something similar to the ALICE analysis train model, where organized processing passes executing user code are launched with a predictable schedule. By 2015, CMS will have one of the two in operations, which should make more predictable use of the computing for production of analysis group data formats.

CMS expects several improvements and consolidations in the computing services during 2013 and 2014 in preparation for the next run. The largest improvements expected are in the transition to multi-core workflows and the use of opportunistic computing and additional architectures. We expect to take advantage of the pilot system to facilitate the transition to multi-core scheduling. If the pilot system is given groups of processors or entire nodes, the system can schedule the resources as if they were batch nodes. It will take a mix of multi-core and single core jobs to make the most efficient use of the available resources. The plan is to go to multi-core scheduling only, by March 2014.

Analysis will probably always be single core because it is not possible to trust the users to properly manage multiple threads. It is therefore important to move as many analysis activities as possible to organized production (e.g. group ntuple productions).

The duration of the batch slot has to be increased when we go to multi-core scheduling. We currently estimate that we would be losing 50% efficiency when the pilot lifetime is 48h and scheduling single core jobs, but this would go down to ~20% if queue went to 72 hours. These changes require negotiation with the sites. Another more ambitious possibility is extending the lease for a pilot. This is not possible with current batch systems but can be considered if abandoning them (e.g. on clouds). In any case we would benefit from knowing the duration of the resource allocation.

In addition to moving to multi-core, CMS sees a benefit to increasing the number of architectures we are able to use. The commissioning phase and run 1 were something of an artificial period with homogeneity centred on Scientific Linux (Enterprise Linux). Previous experiments have had many more platforms and there is growth in a variety of areas. Scheduling should be able to cope with heterogeneous resources: multiple architectures (e.g. ARM 64bit microserver farms, Xeon Phi's or other X86 variants, GPGPU's, PowerPC farms (Bluegene machines)).

In addition to multiple architectures CMS has a strategy to exploit opportunistic resources in general, not only in cases like increasing an existing site or configuring a new site like a standard CMS site. The resource type should match the CMS requirements (memory, disk, ...). A lot of development and adoption of services in needed to take resources with no WLCG services and integrate them dynamically into the distributed computing system. The software distributed is provided by CVMFS but mounted in user space and translated with a product called “Parrot”. The role of the CE is replaced with a bridge to the local batch

system with a tool called BOSCO. Data serving for opportunistic computing relies heavily on remote IO and Xrootd, which in term assumes reasonable networking to the opportunistic resources.

CMS should be able to cope with pre-emption. Initially treat it as a failure and measure the impact on operations. In the long term we may use checkpointing (breaking job in small chunks -> bookkeeping). Note that this would not be full checkpointing, rather something to be implemented inside the application. The decision whether to implement application level checkpointing is a balance between increased efficiency and manpower needed to implement it. Use the use case of the access to the HLT farm in the interfill periods as a parameter to determine whether implementing checkpointing. We should try to understand the operation mode and base a software solution on that use case. Otherwise stay with normal (~1 h) multicore jobs.

We will declare if a workflow is "opportunistic-compatible". This would produce shorter jobs (but then they are shorter everywhere). The workflow tools may be instrumented to restart execution from an intermediate point when resubmitted in case there is information about a previously partial execution. Job splitting depending on whether you run on opportunistic resource: opportunistic would write out one file per lumi and continues processing, in parallel thread that stages out the individual files for what was missed.

The efficient use of opportunistic computing is an important goal for CMS because it can provide significant increases in capacity with low cost. CMS had good experience in the spring of 2013 with the use of the San Diego Supercomputer Centre for a month, where 8000 cores were provided to process a parked dataset. This increased the Tier-1 processing capacity by 40% when it was available. All the improvements needed for opportunistic computing also have the potential for allowing the use of the HLT during the year. The HLT has periods between the fills and during development when the farm is available for processing, but the jobs need to vacate with short notice, while trying to minimize the loss of useful work. We expect the work in opportunistic computing will also help us make use of the HLT, but it may be the second year of Run2 before we can continuously use the HLT during the interfill periods.

Another significant addition to CMS distributed computing resources is the ability to provision clouds. The work done to utilize the HLT resources is directly applicable to other resources provisioned with OpenStack. We expect that these techniques will be used to connect to the CERN Agile Infrastructure resources, academic cloud resources, other contributed clouds, and potentially even commercial clouds if the cost models become more competitive. The pilot infrastructure through the Glide-In WMS provides a clean interface to the production system. The goal of CMS is to be able to double the simulation capacity of the experiment for the period of a month using cloud provisioned resources. In 2013 this would be the ability to utilize 30k simultaneously running cores, and a larger number after the increases expected in 2015. It is not clear whether we will be contributed or could afford to purchase this level of increase, but the production and provisioning system are targeted for this scale. A factor of 2 increase for a month would less than a 10% increase in the total

averaged over the year, but would be a noticeable change for the user community.

We will also consider volunteer computing. Small scale computing provided to use resources that would otherwise be lost. As an example, BOINC is being integrated into condor. This type of computing is probably only useful for specific workflows, but it has the potential for being large and free except for the obvious development and operations costs.

2.3.9 CMS Deletion Policy

During Run1 CMS had 2 primary deletion techniques: small scale deletions automatically for a defined window of events and large scale organized deletion campaigns. The Express data is deleted completely after 30 days on disk storage at CERN. Any event in the Express stream is also persistently stored in one of the primary datasets, and Express is primarily used for initial calibration and monitoring. The data deletion campaigns are large scale targeted programs for clean up. During Run1 for simulation the RAW formats were deleted first, followed by the RECO formats. AOD tended only to be removed if the samples were found to be wrong. In Run2, the two commonly deleted formats will not be persistently written to tape, which should simplify the deletion.

2.3.10 Differences for Heavy Ion (or p-Pb)

The Heavy Ion program is very similar to the proton-proton program with the primary difference being the number of sites used. The heavy ion events are reprocessed using a dedicated Tier-2 facility in Vanderbilt. This had been a distinguishing feature, but this functionality is also expected in the pp program and the site boundaries are reduced. The Heavy Ion program makes use of only remote archival facility at FNAL plus CERN and has been a successful demonstrator of the ability to process efficiently using archival services at a distance. The Heavy Ion program makes use of roughly 5 Tier-2 computing facilities for analysis, plus some capacity at CERN. The overall scale in terms of number of total data taking time and number of sites used is about 10% of the proton-proton program.

2.3.11 Non-event data

CMS expects only a few significant changes in the non-event data moving in 2015. Oracle will continue to be used for the master copy of conditions, but conditions will be served to applications through the use of Frontier caches both on the online and offline farms. The current hierarchical structure of failover has served the experiment well.

No changes are expected in the logical to physical file name resolution. The Trivial File Catalogue (TFC) has good performance and scalability, and has facilitated the transition to xrootd for wide area access to data. No changes are foreseen for the second run.

The one significant change proposed is the full adoption of CVMFS for software distribution. By the spring of 2014 CMS will expect that all sites use CVMFS to distribute the common software. Three modes of operations will be supported

- Native CVMFS clients,

- CVMFS over NFS,
- CVMFS in opportunistic mode through the use of Parrot.

The third option can be used to enable CVMFS from a non-privileged application. The service is brought up dynamically in user space.

2.4 LHCb

2.4.1 LHCb Data model

The LHCb data model is illustrated in Figure 7, which shows the various data formats and the processing stages between them.

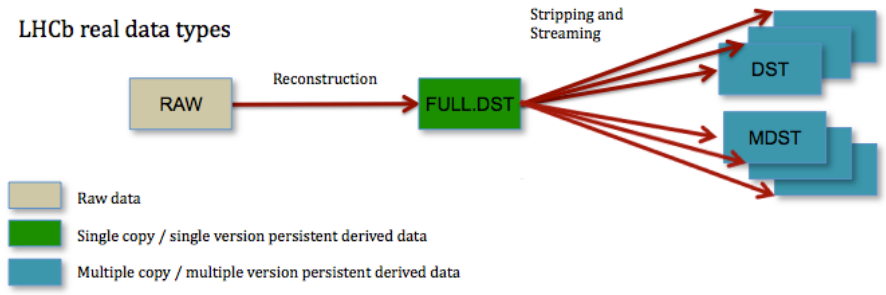


Figure 7: The LHCb Data model

LHCb prompt reconstruction and reprocessing passes are identical. In a first step the RAW data is reconstructed to produce a FULL.DST, which can be used as input for all further data reduction activities. The second step consists of running an application ("Stripping") which selects events for physics analysis or calibration; several hundreds of independent stripping lines are executed, each of which is associated to one of a dozen output streams; depending on the analysis to be performed, a stream is either in DST format or MicroDST (MDST). Corresponding to each MDST there will also be a single copy of a DST (called ALL.DST), containing all events corresponding to MDST streams, and from which the MDST can easily and quickly be re-derived. This is intended to encourage migration to the use of MDSTs, by providing an "insurance" if and when additional information is found to be required during analysis.

Since the stripping step is independent of the reconstruction step, a re-stripping can be performed starting from the FULL.DST. Both reconstruction and stripping activities are organised and scheduled centrally; physics groups only have access to the stripping output datasets. Table 8 lists the various data formats.

Table 8: LHCb data formats

RAW	Raw data: all events passing HLT including luminosity 'nano' events used for the luminosity measurement. Input to reconstruction (prompt or reprocessing)
FULL.DST	Complete reconstruction output for all physics events (i.e. excluding luminosity events), plus a copy of the RawEvent. Self-contained, includes measurement of luminosity for the file. Input to stripping (could also be used for reprocessing, needs validation, particularly for luminosity). Persistent (tape only, one copy), to allow restripping.
DST	Output of stripping: events selected by physics criteria, complete copy of reconstructed event plus particle decay tree(s) that triggered

	selection. Self-contained, input to user analysis. Persistent, multiple copies on disk.
MDST	MicroDST, same event model as DST, but containing only subset of event (tracks,PID) that triggered the selection, and minimal Raw data (mainly trigger information). Self-contained, input to user analysis. Content defined on per stream basis. Persistent, multiple copies on disk.

For simulated data the data flows are illustrated in Figure 8.

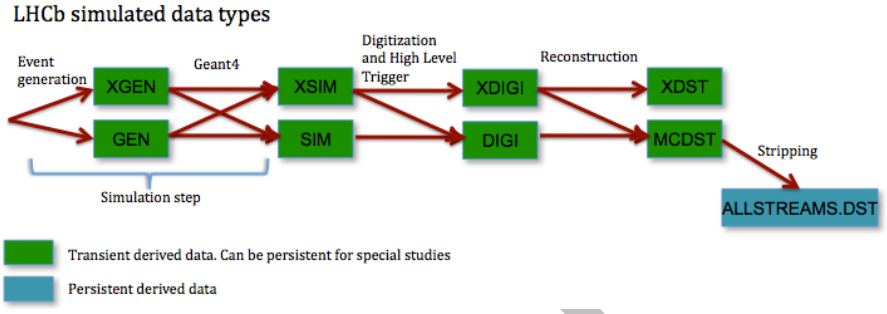


Figure 8: LHCb simulated data processing

The LHCb simulation consists of a number of steps that can be executed in sequence in a single job, using intermediate transient files to transmit data between the steps, but can also be executed independently. Physics analysis normally takes place only on the ALLSTREAMS.DST files, which are the only files saved during massive simulation campaigns. Intermediate files can be saved for special studies, for example generator level studies.

The Trigger, Reconstruction and Stripping steps of simulation productions are identical to those applied to real data, except that there is no streaming at the output of the Stripping, all lines are saved to the same ALLSTREAMS.DST file. The Trigger and Stripping steps can be run in 'flagging' mode, whereby the trigger and stripping decisions are written to the output file but without rejecting events that fail to trigger, or in 'rejection' mode, in which case only events passing the trigger and stripping are written out; the latter mode is typically used when generating large event samples of minimum bias or generic B events, to reduce the storage requirements. The HepMC event record can be propagated to any of the derived formats, but by default is limited to the GEN and XGEN intermediate files to save on storage.

Because of the luminosity levelling, LHCb has rather constant pileup (in time) and spillover (out of time) events overlapping with the event of interest. It is therefore not necessary to simulate many different pileup conditions; in order to simplify the simulation workflow, the pileup and spillover events are generated in the same event generation step as the main event.

It is important to apply to the simulation the same trigger conditions that were applied to the real data. Since the trigger software evolves, it is difficult to

maintain a single version of the software that can run all previous trigger configurations, and difficult for users to know exactly which version of the trigger to run at analysis level. It has therefore been decided to emulate the trigger inside the simulation production workflows; to avoid having different productions for different trigger settings, a scheme is being implemented that permits several different trigger settings to be run in the same job, writing the different trigger decisions to several independent locations in the output files, which can then be selected by the user at analysis time.

It is also planned to mimic the treatment of data, and develop an MC MDST format for simulated data – to both enable easier analysis and reduce storage requirements. This is in development.

Finally we note that we are currently working to reduce the processing time needed for MC events by to ~80% (60%) of current by 2016(2017). This is a very important goal as it strongly affects the overall CPU requirement, which is dominated by MC.

Table 9: LHCb MC data formats

GEN	Event generator output (HepMC format). For generator level studies. Can be used as input to Geant4 step. Not produced in normal productions (Event generation and Geant4 steps are normally merged into a single simulation application)
XGEN	Same as GEN, plus copy of HepMC into LHCb event model classes, for use with standard LHCb analysis tools. For special studies only.
SIM	Output of Geant4 in LHCb event model classes (MCHits, MCParticles, MCVertices), plus beam parameters. Same information stored also for spillover events, if generated. Standard output of simulation step, input to digitization, transient.
XSIM	Same as SIM, but including HepMC record.
DIGI	Simulated RawEvent (same format as for real data), plus copy of MCVertices and MCParticles (only for main event, not spillover) and association tables to allow navigation from RawEvent digitized hits to the MCParticles responsible for them. Standard output of digitization and trigger steps, transient
XDIGI	Same as DIGI+SIM combined, plus association tables to allow navigation from RawEvent to MCHits. For special studies only, can be used to rerun the digitization
MCDST	Same as real data FULL.DST, plus copy of MCVertices and MCParticles (only for main event, not spillover) and association tables to allow navigation from reconstructed objects (e.g. tracks) to the MCParticles responsible for them. Standard output of reconstruction step, transient

XDST	Same as MCDST+XDIGI combined. For special studies only, can be used to rerun the digitization and/or reconstruction
ALLSTREAMS.DST	Same as MCDST for events selected by the stripping, plus particle decay tree(s) that triggered selection. Input to user analysis. Persistent, multiple copies on disk.

Notes:

1. All LHCb datasets are stored as ROOT files, except for RAW data (stored in a home grown, highly compressed, 'MDF' format)
2. All persistent data types are aggressively compressed using LZMA:6 compression in ROOT. This is optimised for disk space and sequential read access on local files. Work is needed to optimise remote I/O and random access if these become frequent use cases.

2.4.2 Storage Classes

LHCb specifies the following storage classes. Unless specified otherwise, providers must guarantee availability and reliability according to the metrics defined for Tier1 sites in the WLCG MOU

- RAW (T1D0): LHCb requires a set of centres providing reliable tape services for long term storage of RAW data. A given site should ideally provide sufficient capacity to record at least 10% of the LHCb RAW data. Write bandwidth should match the site's share of the RAW data output rate. Reading bandwidth is determined by the rate needed to reprocess a given year's data in less than two months. Note however that in the Run2 computing model it is not foreseen to reprocess large quantities of RAW data before the beginning of LS2.
- RDST (T1D0): Storage of FULL.DST output of the real data reconstruction. The FULL.DST dataset is approximately 1.8 times the size of the corresponding RAW dataset. A given site should ideally provide at least 10% of the required capacity. The FULL.DST is accessed periodically (2-3 times per year) during restripping campaigns. This puts quite stringent requirements on the bandwidth out of tape to allow recall of a given year's dataset within a few weeks, although this can be mitigated if we pre-stage ahead of the campaign if space permits.
- DST (T0D1): Storage of DST and MDST for end-user analysis. Sites providing this category of storage must also make adequate CPU provision for analysis. The site's share of this category of disk within the LHCb total pledge should match the site's share of the total CPU available to LHCb for end-user analysis workflows. A Tier1 site qualifies for this service if it provides at least 5% of the LHCb disk request. As of 2013, such disk provision can also be made at any Tier2 site that provides at least 300 TB and that satisfies certain reliability requirements. Such sites are referred to as T2-D. For countries hosting a Tier1 we leave it up to the countries to decide on the most effective policy for allocating the total Tier1+Tier2 disk pledge, for example the Tier2 disk share could also be provided at the Tier1.
- MC-DST (T0D1): Storage of ALLSTREAMS.DST for end-user analysis. Otherwise identical to DST storage.

- ARCHIVE (T1D0): Archive copy of all official analysis datasets (DST MDST, ALLSTREAMS.DST). Produced at the same time as the corresponding T0D1 dataset. In the current model, access of these tape archives is very rare, mainly for data preservation purposes. Such archives do not require highly performing tape drives (access is not time critical) but they do need to be reliable, so ideally at least two tape copies should be kept; since this has a sizeable cost, LHCb currently keeps just one copy and accepts that a small fraction of the archived data may be lost.
- BUFFER (T0D1): Large disk buffer pools, typically located at Tier0 and Tier1 sites, to which all production jobs upload their output. When the production is running on a site that does not have such storage (e.g. any Tier2 site), the output is uploaded to a specific BUFFER SE that has been associated to that Tier2 in the system configuration. The BUFFER SEs must be sized to hold several weeks' worth of output of jobs running at associated sides. The data written to these buffer disks are kept here until processed further; further processing can consist of a new processing step (e.g. stripping after reconstruction, or merging of many small files into larger files) or a data "transformation" step (e.g. replication to DST and ARCHIVE SEs). When all such processing is completed, the files are removed from buffer storage.
- FAILOVER (T0D1): A failover mechanism exists that allows production jobs to upload their output to any other Tier1 in case the associated Tier1 is not available. The data written to this storage class are automatically moved to the destination storage (DST, MC-DST, BUFFER) when the associated Tier1 comes back online.
- USER (T0D1): Persistent storage for user data, managed by quotas applied by the LHCb data management system.

Note that in the current implementation, the DST, MC-DST, BUFFER and FAILOVER storage classes are defined within a single lhcbdisk space token.

2.4.3 LHCb Event sizes

In LHCb, data event sizes stay tend to constant throughout a fill and across fills, due to luminosity levelling. The event sizes shown in the two parts of Table 8 are those measured in 2012. For 2015 and beyond we expect the event sizes to remain roughly the same: although the size of single interaction events will increase due to increased track multiplicity (13 or 14 TeV centre of mass compared to 8 TeV), and increased spillover (25ns vs. 50ns. bunch spacing), this will be offset by lower pileup as a result of doubling the number of colliding bunches (25ns vs. 50ns. bunch spacing), even if we tune the luminosity levelling for the maximum instantaneous luminosity allowed by radiation damage considerations ($6x10^{32}$ $cm^{-2}s^{-1}$, compared to $4x10^{32}$ $cm^{-2}s^{-1}$ in 2012). The Tables below also show the typical numbers of copies of each format kept on disk and tape.

In the past the MC event sizes have been ~ 400kB, however following a campaign of work, this has now successfully been reduced to nearer 200kB which makes a significant saving in projected disk space.

Table 8a: LHCb event sizes for different data formats, and number of replicas for real data

Data Occupying T0/T1/T2 Storage – Planned Model (Data)					
	Event Size kB (2012)	Disk replicas of each version		Number of versions, Typical	Tape Copy
		Min allowed	Typical		
RAW	60	0		1	2
FULL.DST	110	0		1	1
DST	120	2	4	2	1
MDST	10	2	4	2	1

Table 8b: LHCb event sizes, and number of replicas for MC

Data Occupying T0/T1/T2 Storage – Planned Model (Simulation)						
	Event Size kB	Sim Events /Data Events	Disk replicas of each version		Number of versions, Typical	Tape Copy
			Min allowed	Typical		
ALLSTREAMS.DST	200	1.5 (after stripping)	2	3	2	1

2.4.4 LHCb Data flows

2.4.4.1 Raw Data Recording

All RAW data from the pit is transferred to CERN Castor and written to CERN Tape (3GB files). A second, distributed, copy of the RAW files is made on Tier1 Tape, shared according to share of total tape pledge. In Run2, it is foreseen to write three distinct and independent RAW data streams:

- "FULL" stream, which will be fully reconstructed and stripped as has been the case in 2010-2012
- "Parked" stream, where the RAW data will be written to tape and not accessed until later (possibly during LS2). A small fraction (<10%) of the events will be duplicated to the FULL stream to allow preliminary studies of these data
- "Turbo" stream, where the analysis will be performed on an MDST-like data format produced directly in the HLT. The corresponding RAW data will be recorded but probably never reconstructed offline. It can eventually be used to regenerate an updated MDST by rerunning the HLT code offline.

2.4.4.2 Prompt Reconstruction

The purpose of prompt reconstruction in LHCb has evolved during the years. In early years, and up to August 2012, all RAW data from the pit was reconstructed

promptly and used for preliminary data analysis. In September 2012 it was decided to use the prompt reconstruction only for data quality monitoring and to prepare calibrations for the first complete reconstruction for physics analysis use (previously referred to as reprocessing); it was sufficient, for this purpose to limit the reconstruction to 30% of the RAW data.

In Run2, it has been decided to apply a new model to the data-processing. Prompt reconstruction for data quality and alignment/calibration purposes will be carried out entirely on the computing facilities at the LHCb experiment site, and are therefore outside the scope of this document. The complete first reconstruction of the new data will be delayed until fully validated calibration and alignment constants become available. Based on the experience gained at the end of Run1, this is expected to take between 2 and 4 weeks. The data will then be reconstructed and stripped. It will not be re-reconstructed ("reprocessed") until much later, most probably not before the end of Run2.

In what follows, "prompt" reconstruction refers to the delayed processing (by 2-4 weeks) described in the previous paragraph. Reprocessing refers to the re-reconstruction foreseen for LS2.

Prompt reconstruction requires access to the RAW data for reading, sufficient BUFFER to temporarily store the FULL.DST before it is picked up by the stripping, and RDST tape storage for the FULL.DST. In order to avoid a large staging pool in front of the RAW data, we plan to copy new RAW data to BUFFER disk, where it will be kept until the prompt reconstruction can pick it up. If we assume a RAW data-taking rate of 25 TB/day, this will require a BUFFER of approximately 400 TB distributed among the sites providing BUFFER storage.

Ideally the reconstruction jobs should run at the sites providing BUFFER storage, but they can also run at any other site that has sufficient networking to download the RAW files (3GB/file) and upload the FULL.DST files (5GB/file) from/to an associated BUFFER SE. Once the reconstruction has uploaded its FULL.DST, the corresponding RAW file can be removed from BUFFER.

2.4.4.3 Stripping

In the LHCb computing model, the FULL.DST is not available for end-user analysis. The Stripping step is a data reduction step that selects events of interest for specific physics analyses from the FULL.DST files; the selected events are streamed to DST or MDST output files that are made available to end users.

The stripping takes as input FULL.DST files residing on BUFFER storage (typically two files per job). When the stripping runs after a prompt or reprocessing reconstruction, the FULL.DST files are picked up from BUFFER as soon as they are produced: the stripping rate is determined by the rate at which the RAW data can be (re-)reconstructed. In the case of restripping, the FULL.DST already exists on tape and has to be staged to; in this case the stripping rate is determined by the rate at which FULL.DST can be restaged from tape, making it important to have a high bandwidth out of tape in this case.

The stripping consists of several hundred independent "lines", each of which makes a selection of (inclusive or exclusive) final states for specific analyses; events that fire a given line are selected and written out to one of ~13 streams;

the grouping of lines into streams is defined according to likely access patterns and on required data content (DST or MDST).

Each stream is written to a distinct output file, all of which are uploaded to BUFFER storage at the end of the job, at which point the corresponding FULL.DST files can be removed from BUFFER. For each stripping stream, when sufficient files are buffered, a "merging" job merges small files into 5GB files for analysis. The input files are deleted from BUFFER, the output files are replicated to DST storage according to the distribution policy defined in the computing model. In addition, one ARCHIVE (tape) copy is made.

Stripping can be either "full" or "incremental". In the case of full stripping, all the streams are redone with new version of the selection software. This corresponds to a new version of the analysis dataset and, according to the computing model leads to a reduction of the number of disk replicas of the previous versions. An "incremental" stripping uses the same version of the selection software as the corresponding "full" stripping and merely selects additional (new or modified) stripping lines. Operationally, an incremental stripping production is the same as a full stripping (both have to access the complete FULL.DST dataset), the difference being that the number of newly selected events is an order of magnitude smaller.

2.4.4.4 Data Reprocessing

The data reprocessing workflow is identical to the prompt processing workflow, the difference being that the RAW data has to be pre-staged into BUFFER disk from tape. Because of the large amount of resources required for reprocessing (tape staging, CPU for reconstruction and stripping, data handling BUFFER space), the current model only foresees reprocessing during long shutdowns. An important limitation due to finite resources is that the creation of a new version of the FULL.DST by a reprocessing implies deletion of the previous version – it therefore becomes impossible to run any further stripping passes consistent with the previous version of the processing.

A very important evolution of our model is that we no longer plan to perform any re-processing during Run2 periods. All of the Run2 data will only be reprocessed at the beginning of LS2. This is a consequence of delaying the prompt processing until calibrations are available as described above, such that the prompt processing has a very long lifetime.

2.4.4.5 MC Production

Simulation workflows consist of a number of steps (see figure on simulated data types) that are run in sequence in the same job. Because of the amount of CPU required for the event generation and GEANT tracking steps, the number of events produced per job is small (a few hundred), resulting in output files of ~100-250MB. This can be reduced even further if the trigger and/or stripping steps are run in rejection mode, in which case each job may produce at most a handful of events, and can be useful to reduce the storage requirements for systematics studies that require large statistics.

Because they have no input data, simulation workflows can run anywhere, including unpledged resources such as the HLT farm or non-WLCG sites. The

output files are uploaded to BUFFER storage, after which they are merged as for real data and replicated to MC-DST disk storage and ARCHIVE tape. A given production site is associated to one BUFFER SE; this association can easily be reconfigured to load balance the system.

The output format of the simulation permits, in principle, to reprocess the MonteCarlo DSTs with a new reconstruction and/or stripping version. In practice however, reprocessing many hundreds of different productions (for different event types simulated) would be very expensive operationally. Coupled with the fact that the simulation software also improves when moving to one reconstruction version to the next, it has been the policy in LHCb to produce new MonteCarlo samples from scratch when the official reconstruction changes. This is likely to continue in future, in particular considering that reprocessing of real data is not planned before LS2.

2.4.4.6 Group Level analysis

Different analysis groups may define further processing steps after the stripping, for example multi-variate analyses or NTuple productions. These activities are centralised as much as possible, the benefits being that the access to computing resources can be scheduled and the output datasets can be registered in the official LHCb catalogues and replicated following standard replication policies.

2.4.4.7 User Level Analysis

Two major classes of user analysis are considered:

- Analysis of official datasets. Users submit jobs to the Grid using the Ganga framework; Ganga forwards submission requests to Dirac, which prioritises jobs and executes them at the sites where the input data are located. Any output data is uploaded and registered to USER Grid SEs.
- Toy MonteCarlo. The job submission and data registration mechanisms are the same as for data analysis, but the jobs can run anywhere, since they are not restricted to the location of the input data.

2.4.4.8 Access to data by jobs

In all LHCb production workflows, input data files are copied from a grid SE to the local disk of the worker node when a job starts, and output data files are uploaded to a grid SE at the end of the job. It has been found that this model leads to a greater overall processing efficiency, due to the non-negligible probability of failure in opening a remote file by a running job. Production jobs are configured such that all input and output data fits into a 20 GB disk allocation on the worker node.

In data analysis activities (user jobs), with sparse data access on large numbers of 5GB input files, the above model cannot work and data are accessed remotely, in general via the xrootd protocol (though other protocols are also supported by the application software).

2.4.4.9 Locality of data

Because production jobs download their input data from a grid SE, such jobs can in principle run at any of Tier0, Tier1 or Tier2 sites. When sufficient CPU

resources are available at the site holding the data, it is advantageous to run jobs at that site, to minimise WAN traffic, but in cases where a production is CPU limited (e.g. reprocessing) all sites can be used, provided there is sufficient network bandwidth between the worker nodes where the jobs execute and the disk storage where the input data resides. Simulation jobs are a special case: since they have no input data they can run at any site.

Since analysis jobs access directly their input data from storage, without making a local copy, it is more efficient to execute them on the sites with least latency to the storage, which in practice means the sites holding the input data.

2.4.5 LHCb Storage and replication strategy

As has been described, the LHCb analysis is performed on the DST or MDST formats (and corresponding formats for simulation). These are the only datasets that are required to be replicated on disk. Typically we keep on disk the DST and MDST resulting from the latest stripping (or simulation) campaign and, with a reduced number of copies, those resulting from the previous processing campaign. In the current model we keep four copies of the latest processing (three for simulated data) and two copies of the previous processing, which are placed on disk at distinct Tier1 sites in proportion to the available disk and CPU at each site. The tape copy is intended for long term archiving and is in principle never accessed.

In future we expect to have a more granular definition of datasets and to adjust the number of copies in response to dynamically determined predictions of future use. We are participating in an R&D activity together with CMS and the CERN IT-SDC group to develop metrics for such predictions. In such a model it may even be acceptable to keep rarely used datasets only on tape, with scheduled recalls in response to analysis group requests. In common with other experiments we are also investigating strategies for accessing data remotely (a.k.a. storage federations).

Due to a shortage of tape resources, only one archive copy of these derived data is kept. Although in principle the data can be regenerated, this is a very costly operation in terms of both manpower and data processing, so a safer strategy may be needed for long-term data preservation.

The FULL.DST is normally kept only on tape, and is recalled prior to new stripping campaigns. Only the latest version of it is kept, since the analysis model of LHCb imposes that all new analyses (requiring new stripping lines) should be performed on the most recent reconstruction available. Only one tape copy is kept, since individual FULL.DST files can easily be regenerated by re-running the most recent reconstruction on the corresponding RAW files (this is however a manual operation and therefore manpower intensive). It is currently planned to maintain the complete 2015 FULL.DST on disk for as long as resources allow. Small amounts of FULL.DST will also always be kept on disk to allow development and commissioning of new stripping lines.

The RAW data is normally kept only on tape, and is recalled prior to re-processing campaigns. Two copies are kept at distinct sites to protect against tape losses. Small amounts of RAW data are kept on disk to allow development

and commissioning of new reconstruction code or new alignment and calibration strategies.

The replication strategy is given in Tables 8a and 8b. For a given dataset, the LHCb computing model defines how many copies should be available on disk for analysis. The last step of any production workflow replicates the newly produced files over the required number of disk storage elements. All sites that have disk eligible for analysis use (Tier0, Tier1 and certain large Tier2 (known as T2-D)) are considered by the replication transformations.

2.4.6 Anticipated RAW data rates

Rates below are given per second of LHC colliding stable beams. [Note that this means that the instantaneous rate out of HLT farm is lower since it is averaged over periods with no beam, and the instantaneous rate out of stripping is higher since a stripping campaign takes less integrated time than the corresponding data taking.]

2012: The trigger rate was 5kHz leading to (300 MB/s) out of HLT farm.

2015: The changes in beam conditions and in the HLT foreseen for 2015 are described in LHCb-PUB-2013-008. We will run with an output rate of up to 12.5 kHz. If we assume 60kB per event, we expect a RAW data rate of 750 MB/s. The event mix of FULL/PARKED/TURBO/ streams will depend on the offline resources actually available. Current thinking is shown in the Table below. The limiting factor that determines the prompt and parked rates is (i) CPU required to produce MC commensurate with the data and (ii) available disk storage. It is anticipated that through a combination recovery of space, and control of the MC production we can process all data promptly up to at least 2016. In 2017 it may be necessary to park some data - we denote this to be in the range 0-5 kHz of the total of 10 kHz to be fully reconstructed.

Table 10: LHCb trigger rate comparison 2012-2015. 1 Hz means one event per stable-beam-second of running.

Year	Rate Prompt	Rate Parked	Rate Turbo		Output from LHCb
2012	5kHz	0kHz	0kHz		300MB/s
2015	10kHz	0kHz	2.5kHz		750MB/s
2016	10kHz	0kHz	2.5kHz		750MB/s
2017	5-10kHz	0-5kHz	2.5kHz		750MB/s

2.4.7 Summary of schedule for processing/stripping/restriping/incremental stripping

In the description above we introduced the notions of:

- A full “prompt” reconstruction (version m) carried out during the un, but with a short delay to wait for calibrations. This which will be the only reconstruction until LS2.

- A prompt stripping applied at the time of prompt reconstruction. This may evolve for bug fixes and the like, but will be the same version.
- A full restriping applied at the end of the year + incremental strippings applied to the same reconstruction
- A further full restriping + incremental strippings applied a year later
- A final reprocessing + stripping in LS2.

The table below sets out the canonical timing scheme for any given year's worth of data (2015,2016,2017). The intervals are approximate and will vary ~ few months to accommodate scheduling of several years data.

Table 11: LHCb yearly processing schedule

What	Ver.	When
1st "Prompt" reconstruction	R(m)	Prompt + ~4 weeks
1st Full "prompt" Stripping	S(n)	Concurrent w. prompt
2nd Full Re-Stripping	S(n+1)	End of year (EoY)
Incremental stripping	S(n+1)p1	EoY+ 6 months
Incremental stripping	S(n+1)p2	EoY + 10 months
3rd Full Stripping	S(n+2)	EoY +1 year
Incremental stripping	S(n+1)p1	EoY + 18 months
Incremental stripping	S(n+1)p2	EoY + 22 months
2nd Reconstruction	R(m+1)	LS2
4th Full Stripping	S(n+2)	LS2

2.4.8 Differences for Heavy Ion (or p-Pb)

LHCb does not take data in Pb-Pb and does not envisage doing so.

In p-Pb data LHCb has, up to now, run at very low luminosity, data volume is not a problem and processing times are comparable with standard p-p running at 4*10^32. Processing workflows and dataflows are the same as for p-p data (but with different stripping streams)

2.4.9 Non-event data

Dirac databases: These use MySQL; currently these are hosted on private VO boxes, but we plan to move to the MySQL On-demand CERN-IT service. The current DB size exceeds 200GB; the majority of the data is short lived, and we keep only a small fraction, so growth will be slow. However the size will increase considerably when we switch from LFC to DFC (Dirac FC), which will use the same MySQL backend. We also maintain storage for output sandboxes and logs on private voboxes disks.

File Catalogue: LHCb is currently using a central LFC instance located at CERN, with separated read-write and read-only services. The performance of the LFC is adequate for the foreseeable future. However LHCb is considering the

replacement of the LFC by the integrated DIRAC File Catalog (DFC) that is better adapted within the DIRAC Data Management System, as it provides natively functionalities that are not available in the LFC. The migration will hopefully take place during LS1.

Bookkeeping: The full provenance information of all files ever produced by the LHCb production system is recorded in the LHCb Bookkeeping System (BK), based on Oracle. The BK is a service located at CERN and is fully integrated within DIRAC. It is heavily used by all physicists for querying datasets, as well as by the DIRAC production system for retrieving datasets to be processed. Some studies are being done on how to improve the performance of the system for the future, although no major problem is anticipated on the medium term. Novel technologies are being explored for replacing the current Oracle-based warehouse. The access to the BK is based on an API, accessible through a Web interface as well as a GUI. It is also integrated within Ganga, the job submission tool used for physics analysis.

Conditions data base: For the description of the detector and the detector conditions data (alignments, calibrations, etc), LHCb uses a Condition Database (CondDB) based on the CERN IT library COOL with the SQLite and Oracle backends. The Conditions database is divided in partitions depending on the use case:

- DDDB: detector description, data that do not change with the time, but can have multiple versions
- LHCBCOND: detector conditions (alignments, calibrations, etc.), data that vary with time and can have multiple versions
- ONLINE: detector conditions data produced automatically, data that vary with time and have a single version for a given time
- DQ: data quality flags
- SIMCOND: detector conditions for simulation, time varying and multi version replacement for LHCBCOND and ONLINE, used to have a better control on the conditions of the simulation

Until LS1 we have been using Oracle servers at the experimental area and at the main site to synchronize the CondDB partitions between the two sites, but the access to the data has always been via SQLite files (one per partition) containing either the full data set or a snapshot for a specific version. After LS1 we may change the strategy and move to an SQLite-only solution.

Software release and distribution: The CERN Virtual machine File System (CVMFS c.f. http://cernvm.cern.ch/portal/filesystem) is the distribution mechanism for LHCb Production Software. New releases are installed on the CVMFS Head node at CERN, and from there replicated to grid nodes via several layers of cache. This mechanism allows for easy control of the shared software installed all on the Grid (and is easier to manage than the per-site shared software installation areas that were used previously). Users however do not need to use CVMFS to use LHCb software: it is currently released as compressed tar files, and distributed using a LHCb specific Python script. This simple mechanism has been working until now but it is showing its limits. For this reason, LHCb is reviewing its packaging/deployment procedures in cooperation with CERN PH-SFT (that manages external software and generators). The Red-

Hat Package Manager (RPM) could be used instead of LHCb's homegrown system. Prototypes have been developed showing that this would ease software management, and steps are now being taken to ensure the roll out of the new system.

3 Resource Needs & Expected Evolution

In this chapter we describe the anticipated needs of computing and storage resources over the 3 years of the second LHC run. There are a set of common assumptions on which these requirement estimates are based. These are explained in the first section, followed by detailed explanations of the resource needs for each experiment. Finally we summarise the anticipated requirements over the 3 years 2015-17 in a common format.

3.1 General Assumptions

3.1.1 LHC Running time

We assume that the live time of the LHC and experiments will follow a similar pattern to that experienced during the 2010-2013 run; during the first year (2015) the LHC must be commissioned and ramp up in availability; 2016 is a nominal year of running, and 2017, being a year before a long shutdown and presumably when the LHC is running at peak performance may again be available for longer. These assumptions are listed in Table 13 together with assumptions on efficiency and availability, again based on experience in Run 1.

Table 12: Assumptions for LHC pp running

	2015	2016	2017
LHC start date	1/05/2015	01/04/2016	01/04/2017
LHC end date	31/10/2015	31/10/2016	15/12/2017
LHC run days	183	213	258
Fraction of days for physics	0.60	0.70	0.80
LHC efficiency	0.32	0.40	0.40
Approx. running seconds	$3.0\ 10^6$	$5.0\ 10^6$	$7.0\ 10^6$

Assuming typical Pb-Pb or p-Pb running periods in each year as experienced in Run 1, we have the summary shown in Table 13 for assumed running times during each year in Run2.

Table 13:Assumed LHC live time (million seconds/year)

Year	pp (x10^6) sec	A-A (x 10^6) sec	Total (x 10^6) sec
2015	3	0.7	3.7
2016	5	0.7	5.7
2017	7	0.7	7.7
Total	15	2.1	17.1

3.1.2 Assumptions of pileup

ATLAS has in 2012 presented a complete set of arguments that the 50-ns mode of LHC operations at high pileup would cause several issues, not least a very substantial increase in the computing resources required, thus the assumption in this document is that there will be no extended physics data taking at 50-ns and high pileup values and that LHC will quickly move to 25-ns bunch spacing giving more moderate values of pileup. Consequently, in 2015 LHC is assumed to achieve stable operation at the average pile-up $\mu \approx 25$ for the luminosity of $10^{34}cm^{-2}s^{-1}$ at 25-ns bunch spacing. In 2016-2018, the luminosity according to the current LHC scenario could rise up to 1.5 x $10^{34}cm^{-2}s^{-1}$ at 25-ns corresponding to an average pile-up $\mu \approx 40$, with an corresponding increase in reconstruction (and pile-up simulation) CPU times and event sizes.

Ian Bird 4/9/13 4:20 PM
Comment [1]: This is cut from the ATLAS section – can it be reworded to apply to both ATLAS and CMS?

For ATLAS and CMS the reconstruction times are highly dependent on the levels of pileup. Based on what is currently understood the likely LHC running conditions (25 ns bunch spacing, anticipated luminosities), the following assumptions are made for the average pileup anticipated in each year (Table 14).

Table 14: Assumptions for pileup in ATLAS and CMS

	2015	2016	2017
Average pileup	25	40	40

3.1.3 Efficiency for the use of resources

Since the start of the WLCG the resource requests have had some assumptions regarding the efficiency of being able to use resources. Based on empirical observation of the actual efficiencies during the first 3 years of LHC running, and following discussion and agreement with the Computing Resources Scrutiny Group (C-RSG), the values shown below in Table 15 are assumed by all experiments. Note that following recent RRB meetings the efficiency of use of disk has been set to 1.0 to avoid confusion, while we have introduced a tape efficiency to account for the observation that there is some inefficiency due to various factors (somewhat site dependent), including how often repacking of tapes is done (recovering deleted tape file space), and the effects of not always being able to fill tapes.

Table 15: Efficiencies assumed for the use of resources

	CPU		Disk	Tape
	Organised	Analysis		
Efficiency	85%	70%	100%	85%

3.2 Resource Needs and Budgets

The level of resource requests is driven directly by the event trigger rates, for a given efficiency of being able to use those resources. Clearly the desire is to maximise the physics output balancing what is likely to be available in terms of resources.

In deriving the estimates for the resource needs here, we are guided by the following assumptions:

- An assumption that budgets for computing will remain approximately flat in monetary terms;
- That the technology advances anticipated follow the conclusions outlined in the earlier chapter on technology. In that chapter it was concluded that for a fixed cost, we could anticipate an annual growth in performance of approximately 20% for CPU, 15% for disk, and around 15% for tape.

3.3 ALICE

The goal for Run2 is to reach the integrated luminosity of $1nb^{-1}$ of Pb-Pb collisions for which the ALICE scientific program was originally approved. Targeting his objective for Run2 will allow us to extend the reach of several measurements crucial for the understanding of basic properties of the QGP and consolidate preliminary observations from RUN1 data.

The running scenario as presented in this document has been reported to the LHCC in June 2013. The objectives are as follows:

- For Pb-Pb collisions:
  - Reach the target of 1 nb^{-1} integrated luminosity in PbPb for rare triggers
  - Increase the statistics of the unbiased data sample, including minimum bias (MB), centrality triggered events.
- For pp collisions:
  - Collect a reference rare triggers sample with an integrated luminosity comparable to the one of the $1nb^{-1}$ sample in Pb-Pb collisions.
  - Enlarge the statistics of the unbiased data sample, including MB and high multiplicity triggered events.
- For p-Pb collisions:
  - Enlarge the existing data sample, in particular the unbiased events sample (the collision energy is still under discussion).

To reach these objectives ALICE will exploit the approximately 4 fold increase in instant luminosity for Pb-Pb collisions and will benefit from the consolidation of the readout electronics of TPC and TRD allowing us to increase the readout rate by a factor of 2.

The increased data rate in the consolidated system will also increase the demands on the High Level Trigger system. The current architecture of the HLT system is expected to be scalable to the higher event rates. The performance of the GPGPU based TPC tracking algorithm has been demonstrated during Run 1 to meet the requirements of Run 2. The HLT will thus rely on the continued use of GPGPUs, which reduces the number of nodes in the farm.

This will have the effect of doubling the event rate and the data throughput of the entire dataflow including the migration of data to the computing centre. The data throughput to the computing centre will increase up to 10 GB/s.

During LS1 ALICE will upgrade the existing detectors and install additional detectors: the TRD azimuthal coverage will increase from the current 60% to full 100% and a second electromagnetic calorimeter (DCAL) facing in azimuth the existing one will be installed.

We assume that LHC will operate in 2015, 2016 and 2017 as reported in Table 12 and Table 13, i.e., 3.1×10^6 , 5.2×10^6 , 7.1×10^6 , effective seconds respectively of p-p collisions and 0.70×10^6 effective seconds of Pb-Pb or p-Pb collisions,

assuming a LHC availability for physics of 60%, 70% and 80% respectively and a LHC efficiency of 32%, 40% and 40% respectively. The running scenario for RUN2 summarized in Table 16 will allows us to reach the objectives listed earlier in this document. The aggregate DAQ event rate is 500 Hz.

Table 16: RUN2 running scenario

Year	System	Instant Luminosity ($cm^{-2}s^{-1}$)	Interaction rate (kHz)	Running time (s)	# events ($\times 10^9$)
2015	pp unbiased	2×10^{29}	20	3.1×10^{6}	1.5
	Pb-Pb unbiased & rare triggers	10^{27}	8	0.7×10^{6}	0.35
	pp rare triggers	5×10^{30}	500	5.2×10^{6}	2.6
2016	Pb-Pb unbiased & rare triggers	10^{27}	8	0.7×10^{6}	0.35
	pp rare triggers	5×10^{30}	500	7.1×10^{6}	3.4
2017	p-Pb unbiased and rare triggers	$10^{28} - 10^{29}$	20-200	0.7×10^{6}	0.35

The computing model parameters (processing power and data size) have been taken as the average values extracted from the 2012 data processing of p-p and Pb-Pb data. For the resources needed after LS1, estimates are based on the same CPU power for reconstruction and raw event size augmented by 25% to take into account the increase of the track multiplicity due to the higher beams energy and increased pile up. The computing powers needed to process one event are reported in Table 17. The value for Pb-Pb and pPb reconstruction and MC has been increased compared to the values reported in April 2013 as all the events used for the present estimation include TPC data.

The data sizes at the various stages of processing are reported in Table 18.

A factor 4 for raw data compression has been considered. Replication of the reconstructed data is now limited to two instances instead of three as adopted in the previous years.

Table 17: Processing power in kHEPSpec seconds per event

	Reconstruction	Analysis train	End user analysis	Monte Carlo
pp	0.22	0.17	0.01	1.37
PbPb	3.75	2.49	0.17	46.30
pPb	0.71	1.13	0.09	5.98

Table 18: Data sizes in MB/event

	Raw	ESD&AOD	Monte-Carlo
pp	1.05	0.16	0.37
PbPb	7.50	1.55	21.09
pPb	1.63	0.32	1.73

During LS1 we will reprocess the entire set of data collected taking advantage of the best available calibration parameters and the optimal tuning of the reconstruction parameters.

During the same period a major upgrade of the whole offline environment for reconstruction, analysis and simulation is foreseen to improve the software quality and performance. In addition new developments resulting from the R&D program directed toward Upgrade program, including parallelization, vectorization, GPU algorithms, new algorithms, will be implemented. The parts of new environment will gradually become available after 2014. A partial reprocessing of the data will then be performed.

3.3.1 From LS1 to LS2 (2015-2017)

The time profile of the required resources assumes that the heavy-ion runs are scheduled toward the end of the year. Within this scenario the resources required for a given year can be installed during the second semester. It is important that the resources requested in Tier0 are covered allowing us to process the first reconstruction pass of heavy-ion data promptly in 4 months. The share in Tier1s and Tier2s can be further adjusted depending on the pledges, however the sum of the requested resources in Tier1s and Tier2s is essential to allow us processing the data (reconstruction and analysis) and produce the associated Monte-Carlo data within the year following the heavy-ion run. The disk usage has been estimated in a way to store on disk one reconstruction pass with two replica of all data collected between 2015 and 2017 plus a fraction of the associated Monte-Carlo data limited to keep the amount of requested disk storage at a "reasonable" level. New disk storage can be installed any time during a given year and also during the preceding year. Any new disk can be quickly used and will help to process more efficiently analysis tasks. A deficit in pledged disk in Tier1s plus Tier2s could be recovered with an increase of the disk in

Tier0. It is important that the sum of our disk requirements in Tier01, Tier1 and Tier2 are fulfilled.

Resources required in 2013-2017 are listed in Table 21; only resources for RUN2 have been updated.

Table 19: CPU requirements for 2015-2017[1]

	CPU (kHEPSPEC06)			
	Tier0	CAF	Tier1s	Tier2s
2015	130	45.0	120	200
2016	170	45.0	160	240
2017	200	45.0	210	270

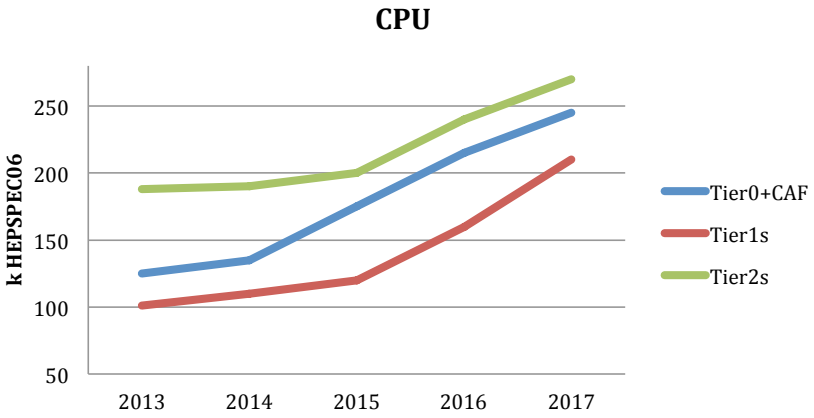


Figure 9: CPU requirement profile. Resources for a given year can be installed during the second semester.

[1] The 2015-2017 values have been updated with respect to the values presented to the RRB in April 2013. Values for 2013-2014 remain unchanged.

Table 20: Tape requirements for 2015-2017[2]

	Tape (PB)	
	Tier0	Tier1
2015	16.2	10.2
2016	21.6	15.6
2017	25.7	19.7

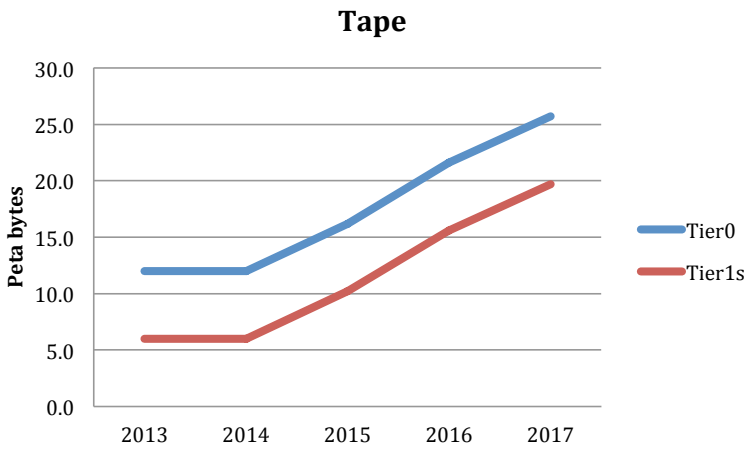


Figure 10: Tape requirement profile. Resources for a given year can be installed at the beginning of the following year.

Table 21: Disk requirements for 2013-2017[3]

	Disk (PB)			
	Tier0	CAF	Tier1s[1)]	Tier2s
2015	11.2	0.34	15.4	22.1
2016	13.4	0.44	18.6	26.8
2017	14.7	0.54	21.8	31.4

1) Excluding the 2.35 PB of disk buffer in front of the taping system

[2] Same as [1]

[3] Same as [1]

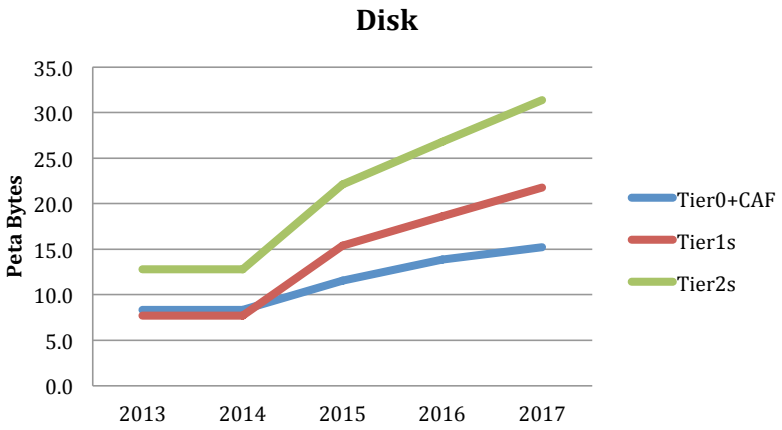


Figure 11: Disk requirement profile. Disks can be installed any time during a given year or during the previous year

3.4 ATLAS

3.4.1 Yearly reprocessing cycles

In Run-2 there is one envisaged full data re-processing from RAW per year on average. In addition, on average two AOD2AOD re-processings of data and Monte-Carlo per year are expected, corresponding to a yearly effective multiplication factor of 2.7 for disk occupancy with respect to the AOD data volume from one (re-)processing. This is based on the assumption that, with the dynamic cleanup present, eventually only the state-of-the-art versions plus the version made at Tier-0 for each period will remain on disk (1 full (re-)processing and the factor 1.7 for the sum of partial AOD2AOD re-processings during the year).

3.4.2 Parked Data

No delayed streams are envisaged for Run-2, which could be re-evaluated after the first year or two of the data taking.

3.4.3 Yearly MC Campaigns

There will be at most two (full Geant4) simulation campaigns per year, one at the beginning of the LHC run with limited statistics and a subsequent full campaign with accurate LHC and detector parameters. This corresponds to an effective multiplication factor of 1.2 w.r.t. the total stored HITS volume from one campaign. The number of Monte-Carlo re-processings is identical to the real data (one full re-processing and on average two AOD2AOD re-processings per year, amounting to the effective multiplication factor for AOD of 2.7).

3.4.4 Re-strippings or group-level

With the introduction of the new analysis model improvements, the total volume of the group analysis (real and simulated) data is estimated to occupy twice the total volume of one AOD replica.

3.4.5 Placement of older data

With the introduction of aggressive dynamic data deletion of pre-placed primary replicas, the total AOD disk volume at the end of a Run-2 data-taking year is supposed to represent the total volume from one re-processing for real and simulated samples, which are retained in the following year in one primary replica at Tier-2. In addition, the group analysis data volume projected to be kept in 2016 is 25% of the total group data volume produced in 2015. Likewise, in 2017, 50% of the 2016 group data volume will be kept (assuming the running conditions remain constant in 2016 and 2017).

3.4.6 Assumptions on running conditions

ATLAS has in 2012 presented a complete set of arguments that the 50-ns mode of LHC operations at high pileup would cause several issues, not least a very substantial increase in the computing resources required, thus the assumption in this document is that there will be no extended physics data taking at 50-ns and high pileup values and that LHC will quickly move to 25-ns bunch spacing giving more moderate values of pileup. Consequently, in 2015 LHC is assumed to

achieve stable operation at the average pile-up μ≈25 for the luminosity of $10^{34}cm^{-2}s^{-1}$ at 25-ns bunch spacing. In 2016-2018, the luminosity according to the current LHC scenario could rise up to 1.5 x $10^{34}cm^{-2}s^{-1}$ at 25-ns corresponding to an average pile-up μ≈40, with an corresponding increase in reconstruction (and pile-up simulation) CPU times and event sizes.

For the purpose of the Run-2 resource estimation, the LHC is assumed to deliver 3,5, and 7 million live seconds of physics collisions in each respective year.

3.4.7 Summary tables of requirements for 2015 – 2018

The tables below summarize the ATLAS resource requirements for Run-2. The requirements are shown to conform to the expected 'flat budget' of cost, which is described by the scaling (modified Moore's law) to resource increase factor of 1.2/year for CPU and 1.15/year for disk and 1.15/year for tape with an uncertainty on the order of 10%. The square brackets [] for the 2015 request show the resource requirements of the ATLAS March 2013 resource request to the Computing Resources Review Board. The values for 2012-2014 shown in the diagrams represent the existing resources and/or validated pledges.

Table 22: Input parameters for ATLAS resource calculations.

LHC and data taking parameters		**2012 pp actual**	**2015 pp μ=25 @ 25 ns**	**2016 pp μ=40 @ 25 ns**	**2017 pp μ=40 @ 25 ns**
Rate [Hz]	Hz	400 + 150 (delayed)	1000	1000	1000
Time [sec]	MSeconds	6.6	3.0	5.0	7.0
Real data	B Events	3.0 + 0.9 (delayed)	3.0	5.0	7.0
Full Simulation	B Events	2.6 (8 TeV) + 0.8 (7 TeV)	2	2	2
Fast Simulation	B Events	1.9 (8TeV) + 1 (7 TeV)	5	5	5
Simulated Data					
Event sizes					
Real RAW	MB	0.8	0.8	1	1
Real ESD	MB	2.4	2.5	2.7	2.7
Real AOD	MB	0.24	0.25	0.35	0.35
Sim HITS	MB	0.9	1	1	1
Sim ESD	MB	3.3	3.5	3.7	3.7
Sim AOD	MB	0.4	0.4	0.55	0.55
Sim RDO	MB	3.3	3.5	3.7	3.7
CPU times per event					
Full sim	HS06 sec	3100	3500	3500	3500
Fast sim	HS06 sec	260	300	300	300

Real recon	HS06 sec	190	190	250	250
Sim recon	HS06 sec	770	500	600	600
AOD2AOD data	HS06 sec	0	19	25	25
AOD2AOD sim	HS06 sec	0	50	60	60
Group analysis	HS06 sec	40	2	3	3
User analysis	HS06 sec	0.4	0.4	0.4	0.4

Table 2: Tier-1 CPU

Tier-1 CPU (kHS06)	***2015***	***2016***	***2017***
Re-processing	38	30	43
Simulation production	154	89	102
Simulation reconstruction	194	245	280
Group (+user) activities	76	187	267
Total	***462 [478]***	***552***	***691***

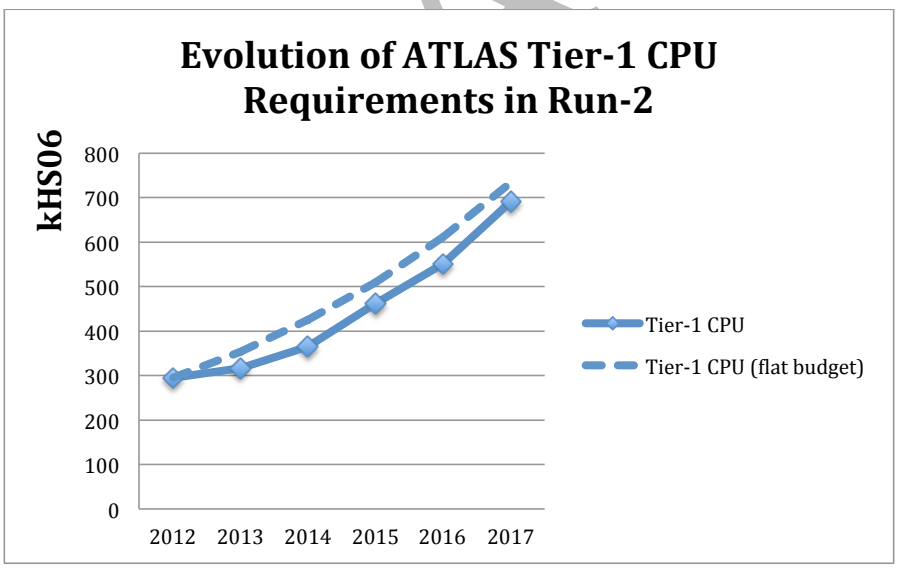


Figure 12: Evolution of ATLAS Tier 1 CPU

Table 3: Tier-1 Disk

Tier-1 Disk (PB)	*2015*	*2016*	*2017*
Current RAW data	2.4	5.0	7.0
Real ESD+AOD+DPD data	5.6	7.9	11.1
Simulated RAW+ESD+AOD+DPD data	9.2	11.4	11.4
Calibration and alignment outputs	0.3	0.3	0.3
Group data	7.5	8.0	10.4
User data (scratch)	2.0	2.0	2.0
Cosmics	0.2	0.2	0.2
Processing and I/O buffers	3.0	3.0	3.0
Dynamic data buffers (30%)	9.0	10.9	12.6
Total	***39 [47]***	***49***	***58***

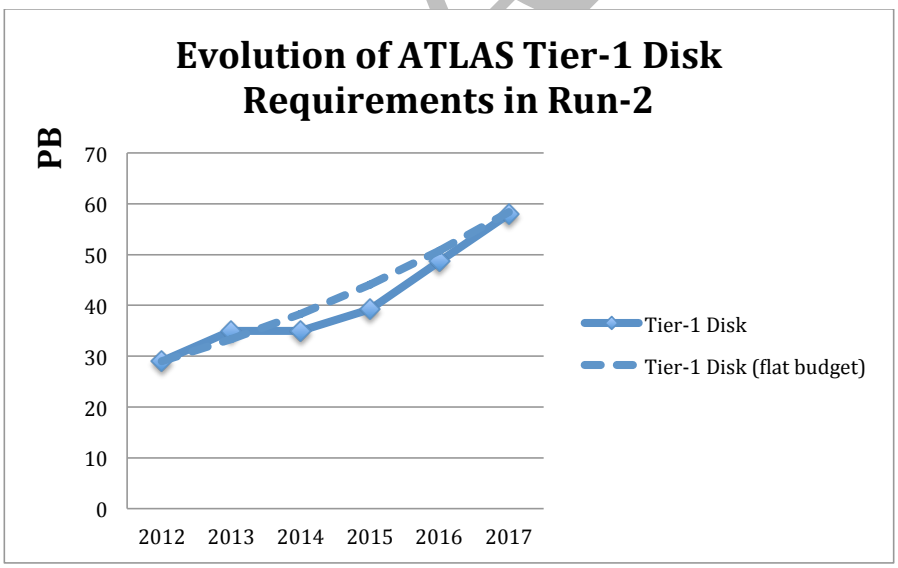


Figure 13: Evolution of ATLAS Tier 1 Disk

Table 4: Tier-1 Tape

Tier-1 Tape (PB) Cumulative	*2015*	*2016*	*2017*
Real RAW+AOD+DPD data	17	26	39
Cosmics and other data	4	4	4
Group + User	7	8	9
Simulated HITS+AOD data	37	46	56
Total	***65 [74]***	***84***	***108***

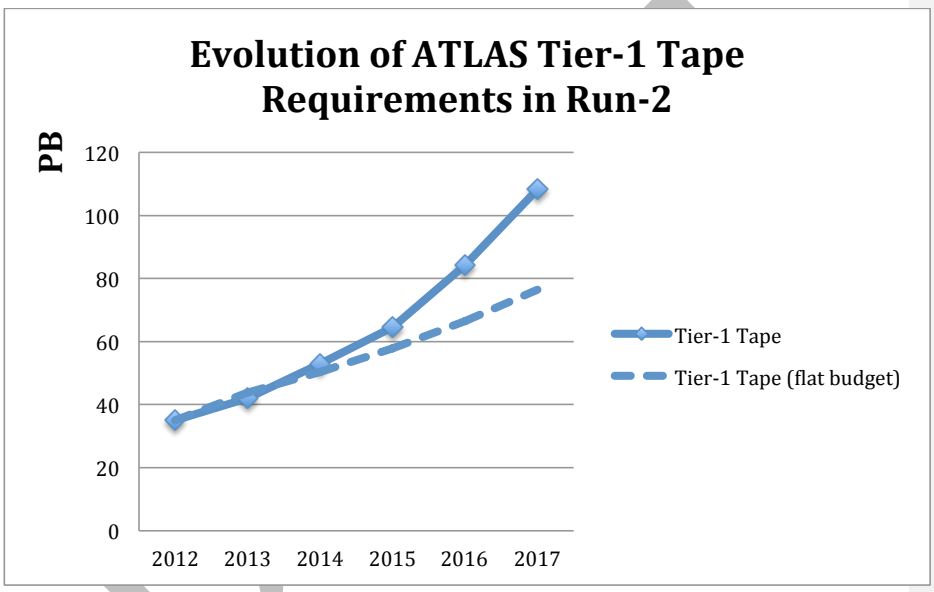


Figure 14: Evolution of ATLAS Tier 1 Tape

Table 5: Tier-2 CPU

Tier-2 CPU (kHS06)	*2015*	*2016*	*2017*
Re-processing	20	33	47
Simulation production	338	347	396
Simulation reconstruction	77	61	70
Group + User activities	96	166	219
Total	***530 [522]***	***608***	***732***

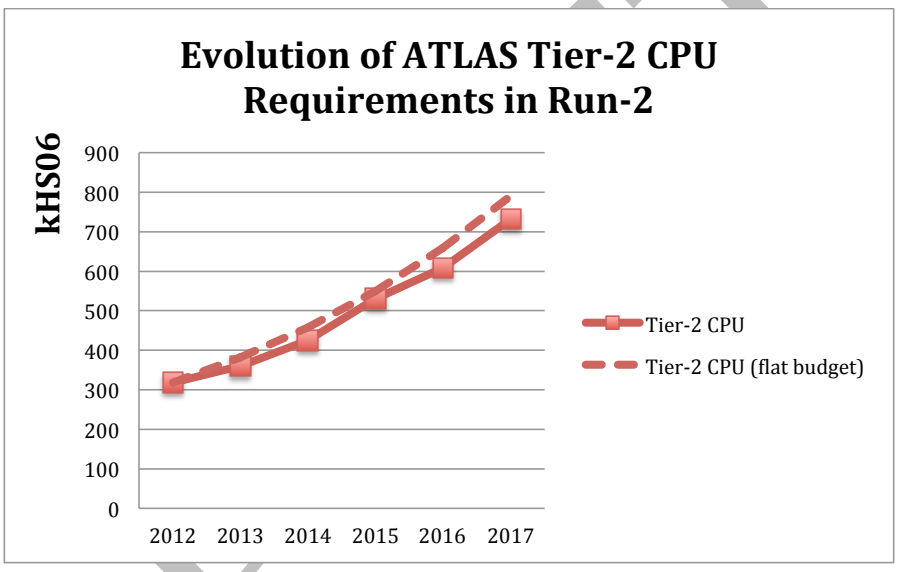


Figure 15: Evolution of ATLAS Tier 2 CPU

Table 6: Tier-2 Disk

Tier-2 Disk (PB)	*2015*	*2016*	*2017*
Real AOD+DPD data	4.1	6.3	10.6
Simulated HITS+RDO+ESD+AOD	10.6	16.6	21.6
Calibration and alignment outputs	0.2	0.2	0.2
Group data	20.4	29.3	41.6
User data (scratch)	4.0	4.0	4.0
Processing and I/O buffers	3.0	3.0	3.0
Dynamic data buffers (30%)	12.7	15.3	16.8
Total	***55 [65]***	***75***	***98***

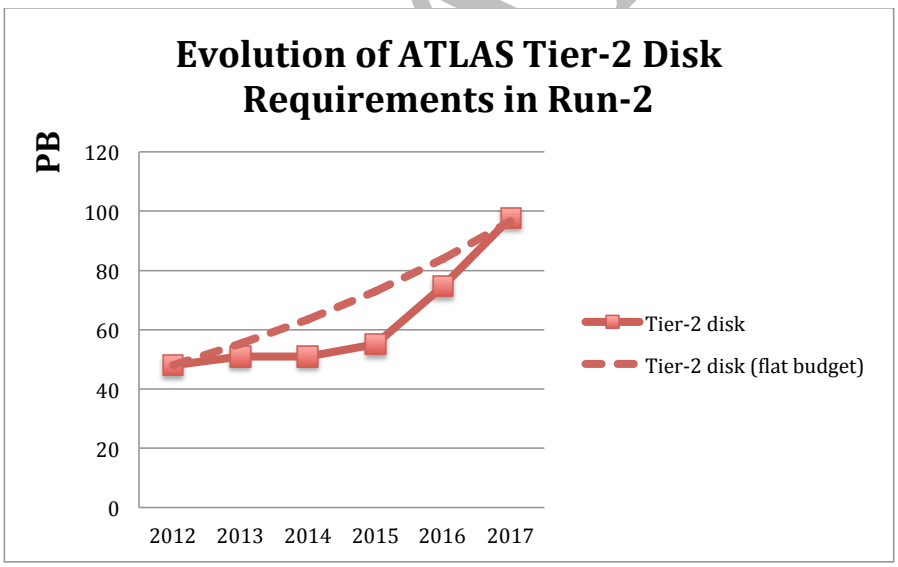


Figure 16: Evolution of ATLAS Tier 2 Disk

Table 7: CERN CPU

CERN CPU (kHS06)	*2015*	*2016*	*2016*
CERN CPU Total	***205 [240]***	***257***	***273***
Tier-0 subtotal	***156***	***199***	***199***
T0: Full reconstruction	133	175	175
T0: Partial processing and validation	12	12	12
T0:Merging and monitoring	4	5	5
T0: Automatic calibration	5	5	5
T0: Servers	2	2	2
CAF subtotal	***49***	***58***	***73***
CAF: Partial reconstruction, debugging and monitoring	13	18	18
CAF: Non-automatic calibrations	4	4	4
CAF: Group activities	15	19	27
CAF: User activities	5	6	13
CAF: Servers	12	12	12

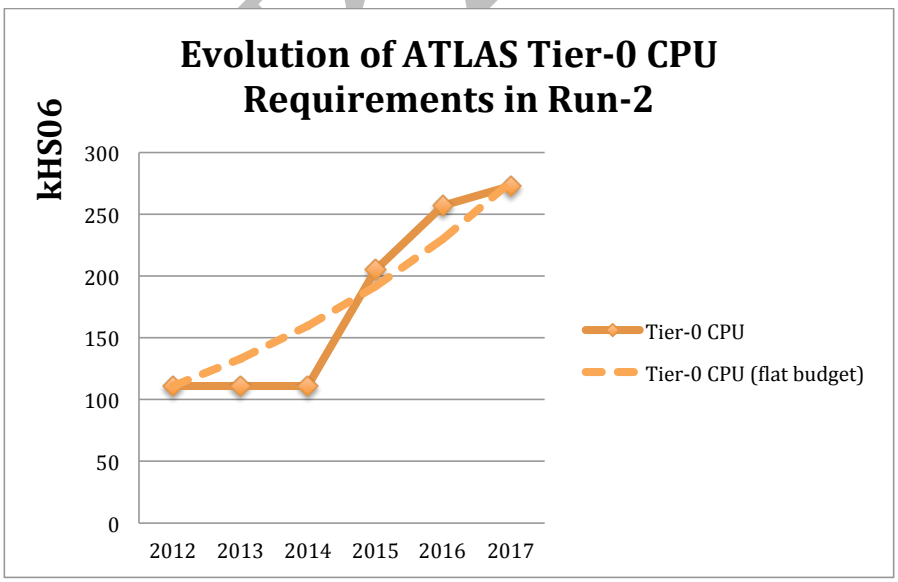


Figure 17: Evolution of ATLAS Tier 0 CPU

Table 8: CERN Disk

CERN Disk (PB)	*2015*	*2016*	*2017*
CERN Disk Total	***14.1 [15.3]***	***17.0***	***19.1***
Tier-0 Disk Subtotal	***3.40***	***3.40***	***3.40***
Buffer for RAW and processed data	3.00	3.00	3.00
Buffers for merging	0.30	0.30	0.30
Tape buffer	0.10	0.10	0.10
CAF Total	***10.7***	***13.6***	***15.7***
CAF: Calibration and alignment	0.5	0.5	0.5
CAF: Derived detector data	2.0	2.8	3.9
CAF: Derived simulated data	6.7	8.8	8.8
CAF: Group data	1.0	1.0	2.0
CAF: User data	0.5	0.5	0.5

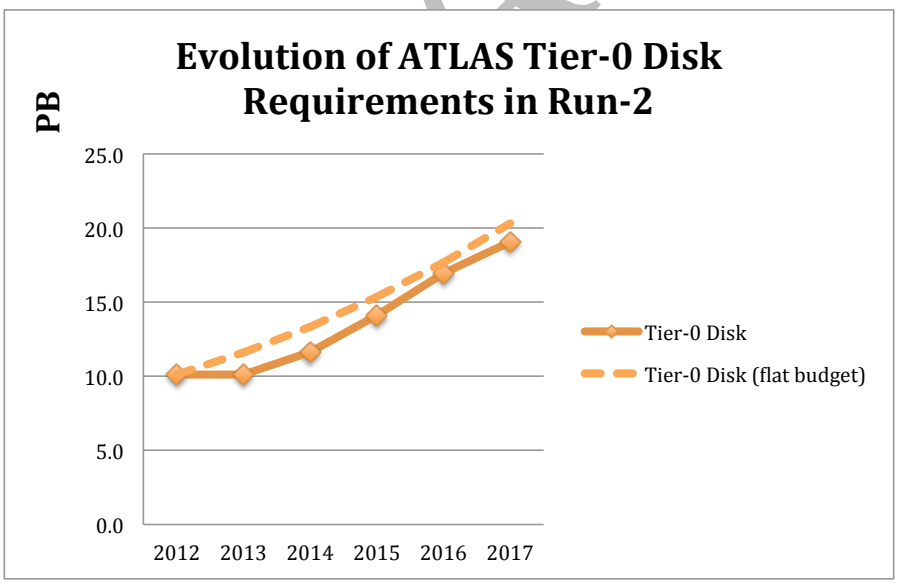


Figure 18: Evolution of ATLAS Tier 0 Disk

Table 9: CERN Tape

CERN Tape (PB) Cumulative	*2015*	*2016*	*2017*
Total	*33 [35]*	*42*	*54*

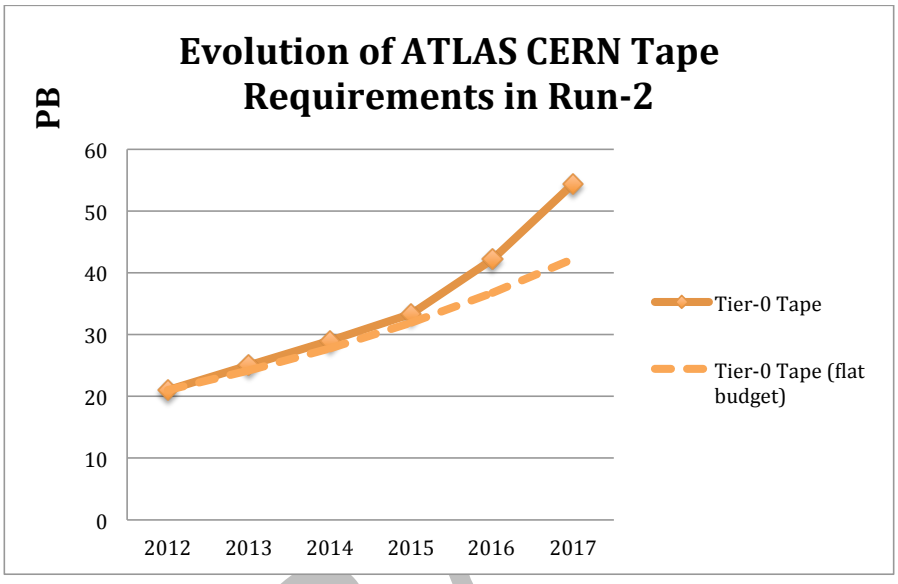


Figure 19: Evolution of ATLAS Tier 0 Tape

3.5 CMS

In 2014 CMS asked for only a small increase in tape capacity to handle new processing and simulation samples. Any additional processing resources were provided by allocating the Tier-0 capacity and using the newly commissioned Higher Level Trigger farm (HLT).

Looking forward to 2015, CMS assume 3Ms LHC run time in 2015 and a DAQ rate of 0.8-1.2 kHz. The increase is required in order to maintain the same physics capability with the foreseen increase in instantaneous luminosity and energy (the latter accounting for a 15% increase in the rate). This will imply a 2.5 increase factor in computing needs with the higher trigger. While luminosity increase will increase the pileup and therefore the reconstruction time, which will imply a further increase factor of 2.5 in computing processing needs. The move to running at 25 ns, together with the increased luminosity, will increase the out of time pileup. Early studies indicated that this could result in another factor of two, but in the planning we assume reworking the reconstruction to mitigate what would otherwise be a further factor of 2.

If CMS maintained the current data model and workflows, it would face an increase in the computing work to be done of a factor of 6 (or 12 with the out of time pile-up effect) while we have asked for less than a factor of 2 increase in the processing resources. In order to address the situation CMS will seek to gain from operation efficiency and access to opportunistic resources , described in the rest of the document, the most important being: the use of the HLT resources whenever possible. Operations gains include the reduction to only one re-reconstruction campaign during the year and the relocation at the Tier-1s (partially freed by the reduction of the re-reconstructions) of 50% of the prompt reconstruction. Optimization of the event formats is also foreseen in order to reduce the storage needs (in favour of CPU capacity).

3.5.1 Yearly re-processing cycles

In Run 2 there is one envisaged full data re-processing from RAW per year on average at the end of the running year when the maximum capacity can be used from the HLT farm. In addition targeted reprocessing passes of individual primary datasets are expected throughout the year, but in total add to a fraction of the full reprocessing pass.

3.5.2 Parked Data

No delayed streams are envisaged for the Run 2, which could be re-evaluated after the first year or two of the data taking.

3.5.3 Yearly MC Campaigns

A simulation campaign equivalent to 1.5 times the number of data events collected is budgeted for 2015, dropping to 1.3 times data in 2016, and 1 times data in 2017. The factor changes with the number of events collected, which is lowest in 2015, and with the expected activities and measurements. CMS expected to produce samples using fast simulation, which has been heavily used for upgrade samples. The difficulty of fast simulation has been to move it to a

transient format. The simulation is fast enough that it is frequently better to reproduce it than to store is persistently.

3.5.4 Placement of data

With the introduction of the Xrootd-based data federation and the use of monitoring of the access level through the popularity service, CMS has reduced the replication factor for data at the Tier-2s. This is reducing the slope of the disk increase in the Tier-2 disk planning,

3.5.5 Summary tables of requirements for 2015 – 2018

The tables below summarize the CMS resource requirements for Run 2. The requirements are shown to conform to the expected 'flat budget' of cost, which with is described by the scaling (modified Moore's law) to resource increase factor of 1.2/year for CPU and 1.15/year for disk, with an uncertainty on the order of 10%.

Table 23: Input parameters for CMS resource calculations.

LHC and data taking parameters		**2012 pp actual**	**2015 pp μ=25**	**2016 pp μ=40**	**2017 pp μ=40**
Rate [Hz]	Hz	400 + 600 (parked)	1000	1000	1000
Time [sec]	MSeconds	6.6	3.0	5.0	7.0
Real data	B Events	6B	3.0	5.0	7.0
Full Simulation	B Events	5	3.0	6	7
Simulated Data					
Event sizes					
RAW Data	MB	0.5	0.65	.95	0.95
RAW Data	MB	0.5	0.65	.95	0.95
RECO Data	MB	0.75	0.8	0.9	0.9
RECO Data	MB	0.75	0.8	0.9	0.9
AOD Data	MB	0.28	0.30	0.35	0.35
AOD Data	MB	0.28	0.30	0.35	0.35
RAW Sim	MB	1.5	1.5	1.5	1.5
RAW Sim	MB	1.5	1.5	1.5	1.5
RECO Sim	MB	0.80	0.85	0.95	0.95
AOD Sim	MB	0.3	0.35	0.40	0.40
CPU times per event					
Full Simulation	HS06 Sec	500	500	500	500
Fast sim	HS06 sec	50	50	50	50
Real recon	HS06 sec	300	525	920	920
SIM RECO	**HS06 sec**	**400**	675	1050	1050

Table 2: Tier-1 CPU

Tier-1 CPU (kHS06)	*2015*	*2016*	*2017*
Re-processing	100	150	200
Simulation production	150	200	225
Simulation reconstruction	50	50	100
Group (+user) activities	0	0	0
Total	***300***	***400***	***525***

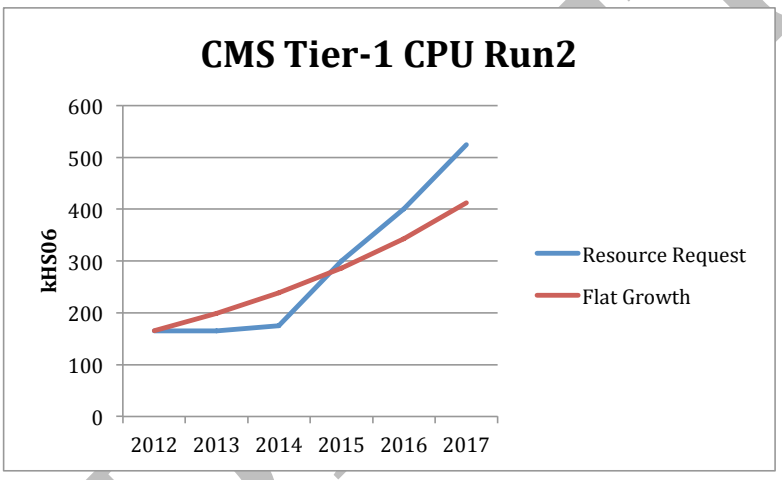


Figure 20: Evolution of CMS Tier 1 CPU

Table 3: Tier-1 Disk

Tier-1 Disk (PB)	*2015*	*2016*	*2017*
Current RAW data	2.0	3.0	4.0
Real RECO+AOD	9.0	11.0	14.0
Simulated RAW+RECO+AOD	8.0	11.0	14.0
Skimming data	3.0	4.0	5.0
User data (scratch)	2.0	2.0	2.0
Dynamic data buffers	2.5	4.0	6.0
Total	***27***	***35***	***45***

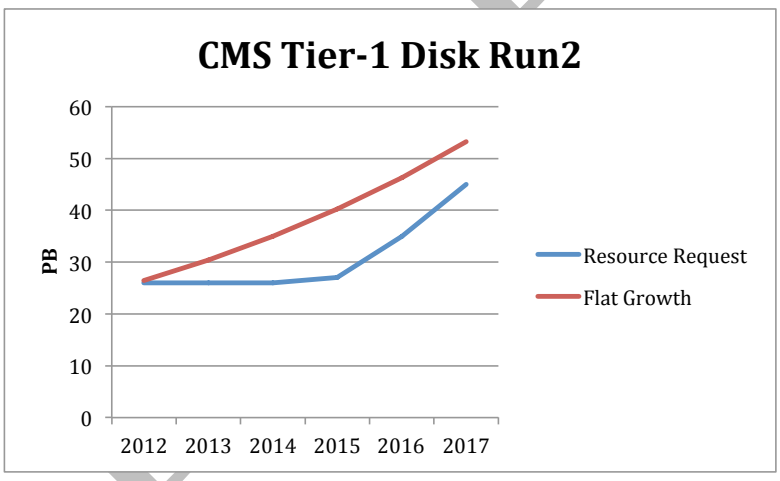


Figure 21: Evolution of CMS Tier 1 Disk

Table 24: Evolution of CMS Tier 1 Tape

Tier-1 Tape (PB) Cumulative	*Run 1 (2014)*	*2015*	*2016*	*2017*
RAW Data	5	9	14	19
RAW Simulation	16	18	20	24
RECO Data and Simulation	19	22	26	34
AOD Data and Simulation	15.5	24.5	40.0	58.0
Total	***55***	***73.5***	***100***	***135***

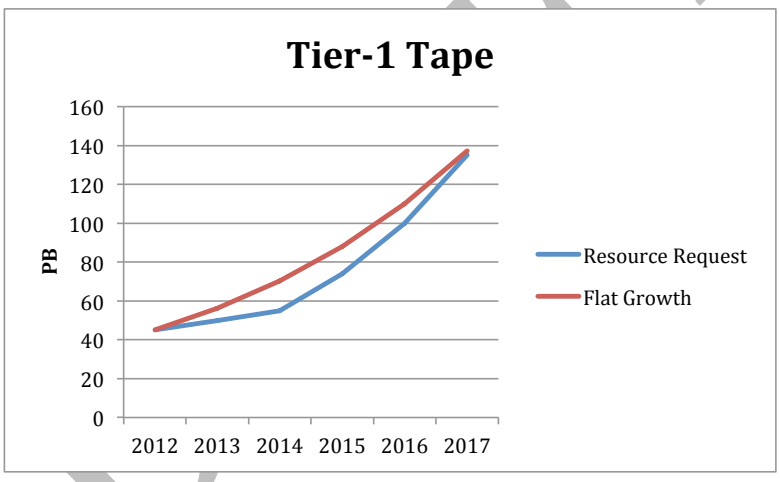


Figure 22: Evolution of CMS Tier 1 Tape

Table 5: Tier-2 CPU

Tier-2 CPU (kHS06)	*2015*	*2016*	*2017*
Analysis	400	550	600
Simulation production	100	150	200
Total	***500***	***700***	***800***

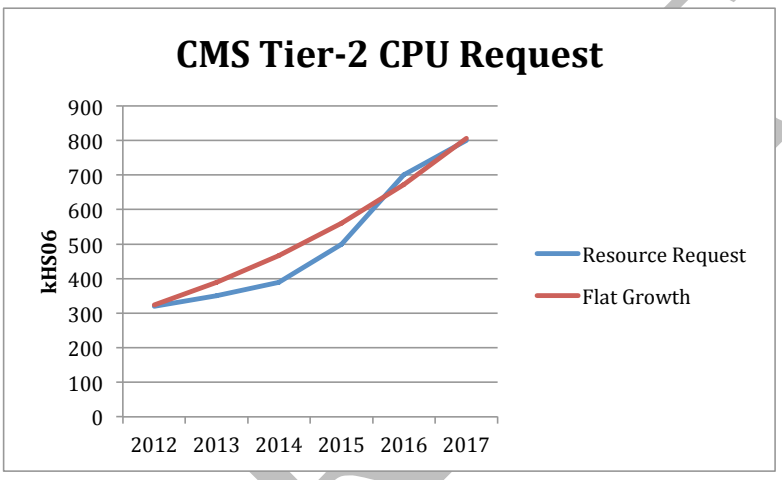


Figure 23: Evolution of CMS Tier 2 CPU

Table 6: Tier-2 Disk

Tier-2 Disk (PB)	*2015*	*2016*	*2017*
Real RECO + AOD	9.0	12.0	15.0
Simulated RECO + AOD	12.0	15.0	17.0
Production Data	2.0	2.0	2.0
User data	8.4	11.0	14.0
Total	***31.4***	***40***	***48***

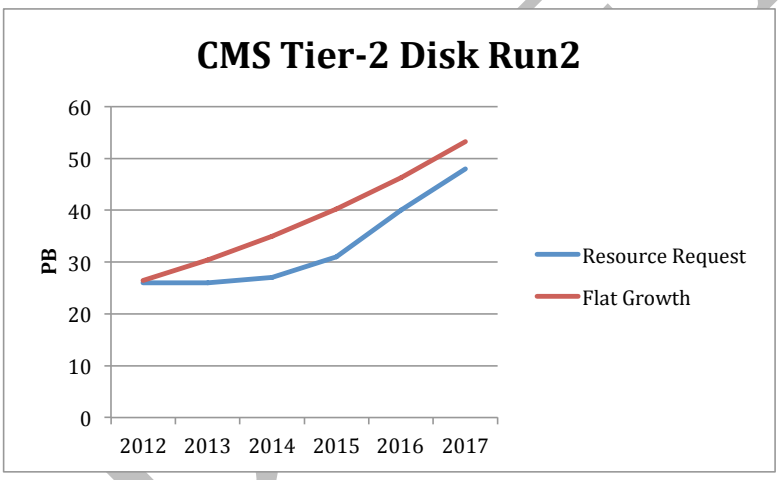


Figure 24: Evolution of CMS Tier 2 Disk

Table 7: CERN CPU

CERN CPU (kHS06)	*Run-1 (2012)*	*2015*	*2016*	*2016*
CERN CPU Total	***135***	***271***	***315***	***365***
Tier-0 subtotal	***121***	***256***	***300***	***350***
T0: Full reconstruction	83	210	246	292
Express	12	17	21	21
T0:Repacking	8	8	10	12
T0: Automatic calibration	6	6	6	6
T0: Servers	12	15	17	19
CAF subtotal	***15***	***15***	***15***	***15***

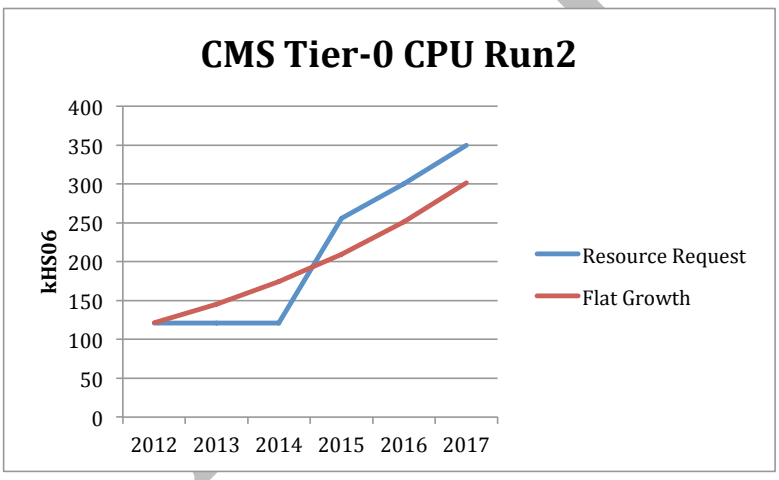


Figure 25: Evolution of CMS Tier 0 CPU

Table 8: CERN Disk

CERN Disk (PB)	*Run 1 (2014)*	*2015*	*2016*	*2017*
CAF and Analysis	*9.0*	*12*	*13*	*14*
Tier-0 and Data Distribution	*0.0*	*3.2*	*3.2*	*3.2*
Total	*9.0*	*15.2*	*16.2*	*17.2*

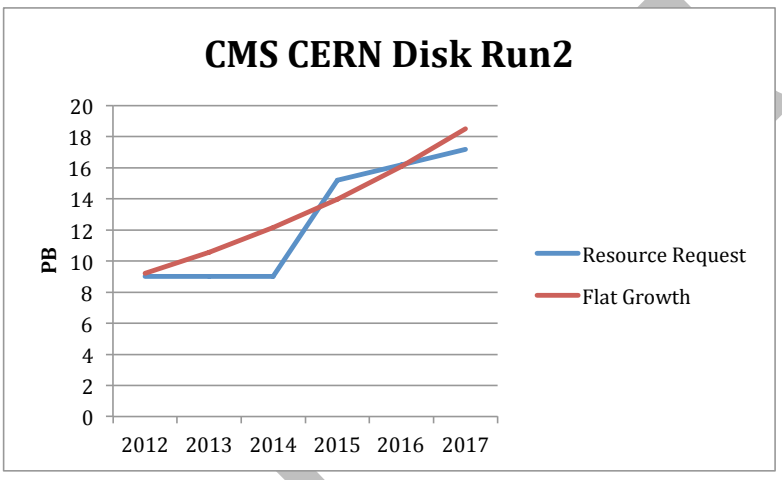


Figure 26: Evolution of CMS Tier 0 Disk

Table 9: CERN Tape

CERN Tape (PB) Cumulative	*Run 1 (2014)*	*2015*	*2016*	*2017*
Total	*26*	*31*	*38*	*50*

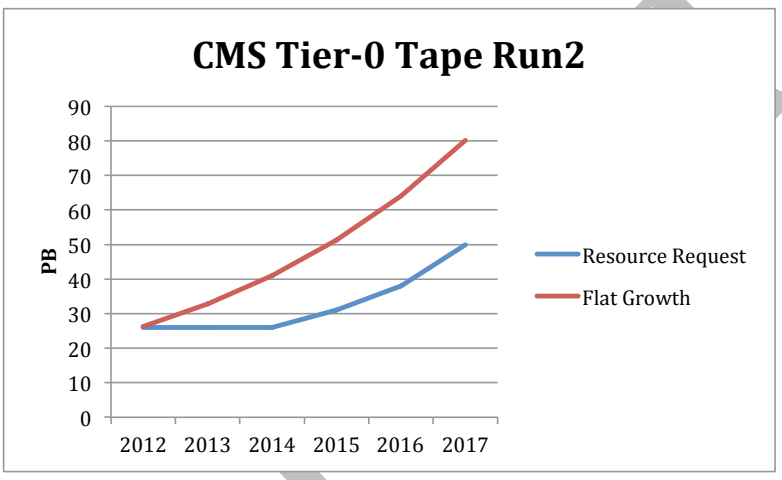


Figure 27: Evolution of CMS Tier 0 Tape

3.6 LHCb

The factors that determine the processing and storage requirements are described in Chapter 2. In summary:

- For 2015 and beyond we expect the event sizes (and therefore processing times) to remain roughly the same as in Run 1 due to the trade off between increased interaction complexity and 25 ns bunch crossing
- Event sizes are given in the relevant table in Chapter 2.
- The trigger rate will be 12.5 kHz (10 kHz to be reconstructed +2.5 kHz "Turbo"). We have introduced the concept of data parking for a fraction of the 10 kHz, but do not expect to need to use it in 2015 or 2016, and we will decide for 2017 based on available resources.
- We assume that the LHC will run with a bunch spacing of 25ns; this is an important parameter for the event size (and therefore computing resources requirements).
- Given currently available tape resources and the expected growth rate in tape requirements, the LHCb data preservation archives consist of a single tape copy, which makes these archives vulnerable to inevitable tape failures. This is clearly an area of concern but we think it is unrealistic to request the additional resource that would be required for a second archive copy.

3.6.1 Data operations

The detailed scheme for data processing/stripping/re-processing for each year of data in 2015-2017 is described in detail and given in the relevant Table in Chapter 2.

3.6.2 Simulation campaigns

For simulation LHCb's model differs from that of ATLAS or CMS, in that the bulk of the MC production is done **after** the year of data-taking, when beam and trigger conditions are known. Once the software is frozen, the simulation runs continuously for up to a year, using idle resources at all Tiers, as well as opportunistic (unpledged) resources.

In 2015 we expect most analyses of 2011-2012 data to be in an advanced state and to have satisfied most of their simulation needs. Simulation efforts are likely to concentrate on further simulations for the LHCb Upgrade studies, and on tuning the simulation to the observed 2015 data-taking conditions.

Since we do not plan to reprocess the 2015 data during the 2015-2016 winter shutdown, we plan to be ready to start a massive MC production for analysis of 2015 data as soon as the 2015 run ends. We expect to have satisfied approximately 50% of the simulation needs for the analysis of 2015 data before the restart of the LHC in spring 2016 (i.e. during the 2015 WLCG accounting period).

The simulation for 2016 is concentrated in the 2016-2017 winter shutdown and continues at lower priority throughout 2017, using any CPU resources not

needed to process the real data. The simulation for 2017 is largely postponed to LS2.

3.6.3 CPU requirements

Table 25 presents, for the different activities, the CPU work estimates for 2015, 2016, 2017. Note that in this table we do not apply any efficiency factors: these are resource requirements assuming 100% efficiency in using the available CPU. The last row shows the power averaged over the year required to provide this work, after applying the standard CPU efficiency factors (85% for organized work, 75% for user analysis).

Table 25: Estimated CPU work needed for the LHCb activities

LHCb CPU Work in WLCG year (kHS06.years)	2015	2016	2017
Prompt Reconstruction	19	31	43
First pass Stripping	8	13	9
Full Restripping	8	20	9
Incremental Restripping	0	4	10
Simulation	134	153	198
User Analysis	17	17	17
Total Work (kHS06.years)	**185**	**238**	**286**
Efficiency corrected average power (kHS06)	**220**	**283**	**339**

The required resources are apportioned between the different Tiers taking into account the computing model constraints and also capacities that are already installed. This results in the requests shown in Table 26. The table also shows resources available to LHCb from sites that do not pledge resources through WLCG.

Table 26: CPU Power requested at the different Tiers

Power (kHS06)	Request 2015	Forecast 2016	Forecast 2017
Tier 0	44	53	63
Tier 1	123	148	177
Tier 2	52	62	74
Total WLCG	**219**	**263**	**315**
HLT farm	10	10	10
Yandex	10	10	10
Total non-WLCG	**20**	**20**	**20**

The request for 2015 has been sized to satisfy entirely with WLCG resources the requirement presented in the Table. This is partly because the contribution from the HLT farm is uncertain (the farm would in any case only be available during the winter shutdown, when many maintenance activities are also required) but also to allow a ramp up, within a constant budget, to the resources required in 2016 and 2017.

3.6.4 Storage requirements

Table 27 presents, for the different data classes, the forecast total disk space usage at the end of the years 2015-2017. This corresponds to the estimated disk space requirement if one assumes 100% efficiency in using the available disk.

Table 27: Breakdown of estimated disk storage usage for different categories of LHCb data

LHCb Disk storage usage forecast (PB)	2015	2016	2017
Stripped Real Data	7.3	13.1	14.7
Simulated Data	8.2	8.8	12.0
User Data	0.9	1.0	1.1
ALL.DST	1.5	1.9	
FULL.DST	3.3		
RAW buffer	0.4	0.5	0.3
Other	0.2	0.2	0.2
Total	**21.7**	**25.4**	**28.2**

Table 28 shows, for the different data classes, the forecast total tape usage at the end of the years 2015-2017 when applying the models described in the previous sections. The numbers include the standard 85% tape efficiency correction, which is probably pessimistic for RAW data that is written sequentially to a dedicated tape class, and never deleted.

Table 28: Breakdown of estimated tape storage usage for different categories of LHCb data

LHCb Tape storage usage forecast (PB)	2015	2016	2017
Raw Data	12.6	21.7	34.5
FULL.DST	8.7	15.2	19.7
ALL.DST	1.8	5.2	7.7
Archive	8.6	11.5	14.7
Total	**31.7**	**53.7**	**76.6**

The disk and tape estimates shown in Table 27 and Table 28 are broken down into fractions to be provided by the different Tiers using the distribution policies described in LHCb-PUB-2013-002. These numbers are shown in Table 29 and Table 30.

As can be seen the increase in disk storage can be managed to fit inside a reasonable growth envelope by adjustments in the details of the processing strategy.

On the other hand, the growth in the tape storage requirement is more challenging but largely incompressible: in Table 28 can see that the major part of the increase is due to RAW data that, if not recorded, is lost. "Parking" of some fraction of this raw data will only reduce by a corresponding fraction the growth rate of tape for FULL.DST (note that Table 28 already assumes parking of 50% of the RAW data in 2017).

Table 29:LHCb disk request for each Tier level. Note, that for countries hosting a Tier 1 we leave it up to the country to decide on the most effective policy for allocating the total Tier 1+Tier 2 disk pledge. For example the Tier 2 share could also be provided at the Tier 1

LHCb Disk (PB)	2015 Request	2016 Forecast	2017 Forecast
Tier0	6.7	8.3	9.5
Tier1	12.5	14.2	15.4
Tier2	2.5	2.9	3.3
Total	**21.7**	**25.5**	**28.3**

Table 30: LHCb Tape requests for each Tier

LHCb Tape (PB)	2015 Request	2016 Forecast	2017 Forecast
Tier0	10.4	15.9	21.6
Tier1	21.3	37.8	55.0
Total	**31.7**	**53.7**	**76.6**

3.7 Summary of resource requirements

The following figures show the summary of the evolution of requests, from 2013 through to 2017. The 2013 data points represent the presently installed pledged capacities, while the points for 2014-17 are the requests discussed above.

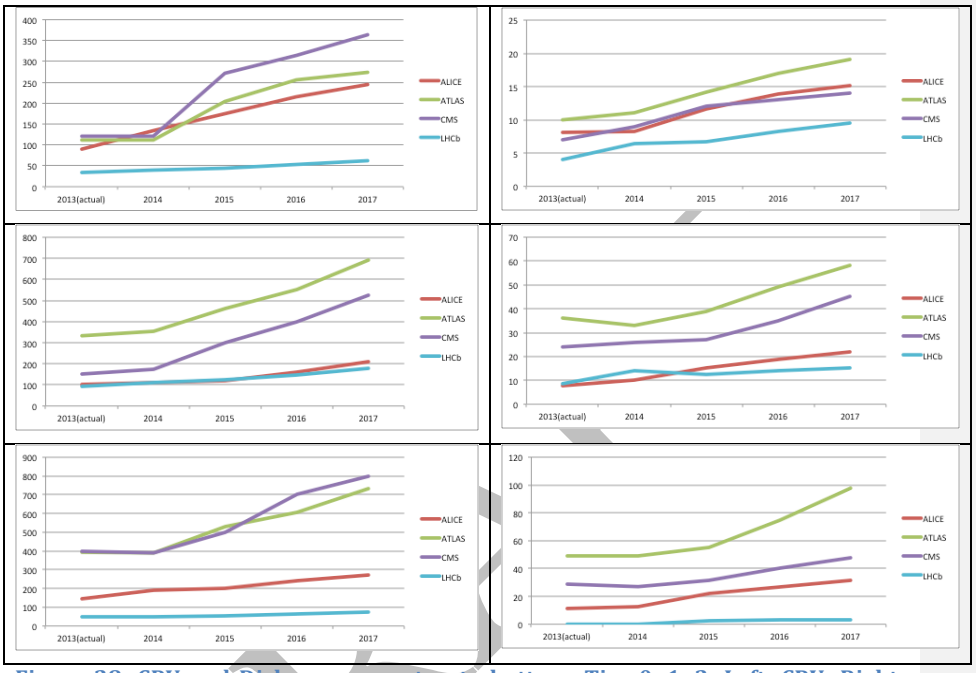


Figure 28: CPU and Disk summary; top to bottom: Tier 0, 1, 2; Left: CPU; Right: disk

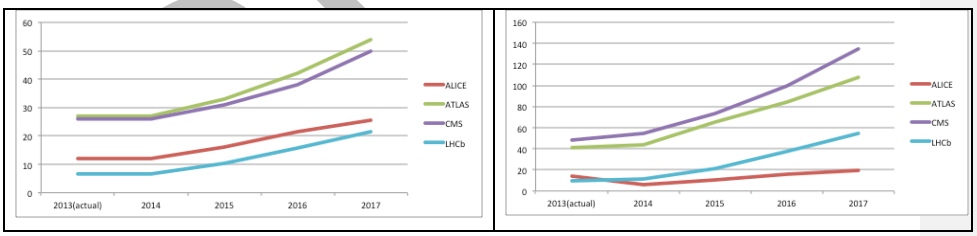


Figure 29: Tape summary; Tier 0 (left), Tier 1 (right)

The following two figures, Figure 30, and Figure 31, show the summary of the evolution of the installed capacities and the new anticipated requirements until 2017. In these figures, the black line is a linear extrapolation of the installed resources from 2008 to 2012, while the blue lines represent the 20% (CPU) and 15% (Disk) yearly growth that is expected from a consideration of technology growth, given flat budgets (see Technology chapter), taking 2013 as the starting point.

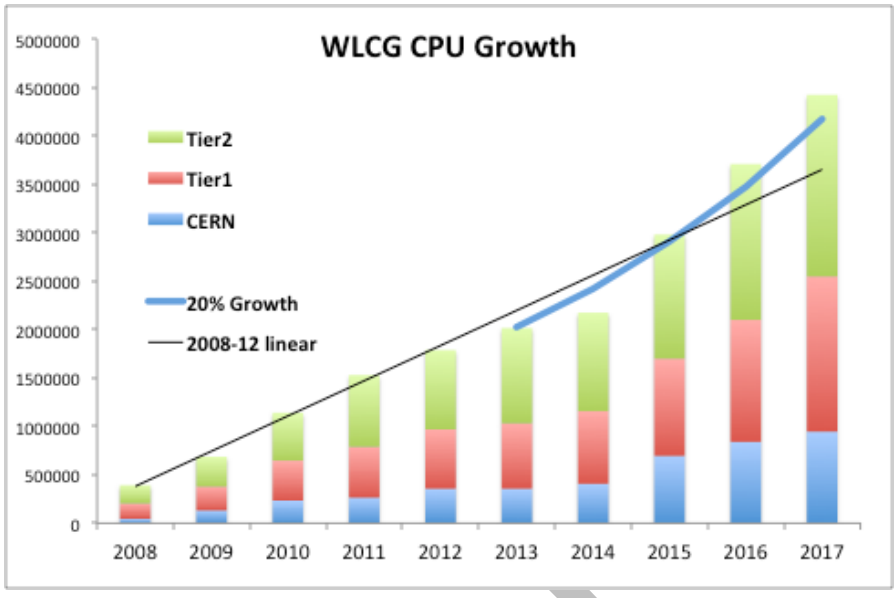


Figure 30: Summary of total CPU requirements

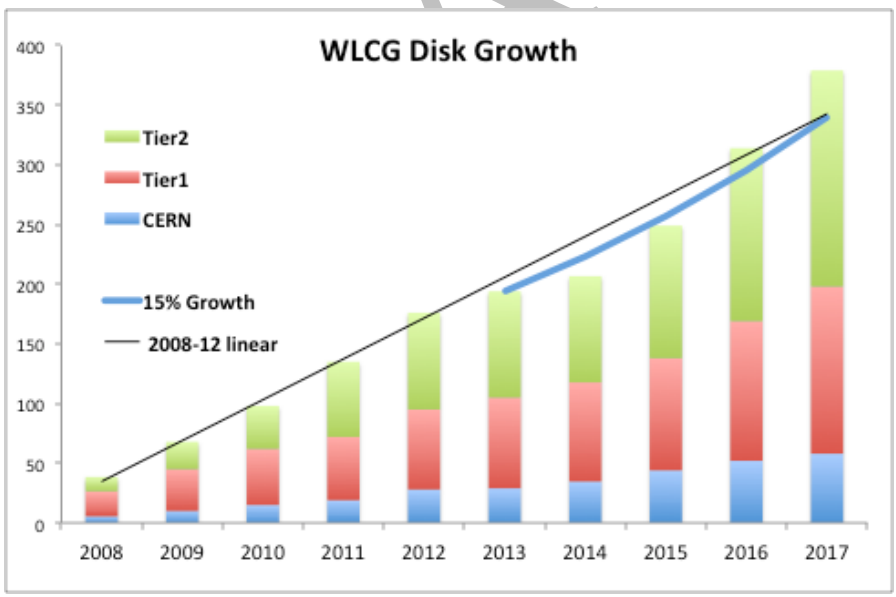


Figure 31: Summary of total disk requirements

4 Technology Evolution

4.1 Processors

The market for products using processors can be essentially separated into three areas, excluding embedded processors, with estimated unit production quantities in 2013:

- Low-end processors based on the ARM design typically used in smartphones (> 1000 million) and tablets (~230 million),
- Mid-range x86 (Intel and AMD) processors used in desktop PCs (~140 million) and notebooks (~200 million),
- High-end processors (of which 98% are x86, the rest being Power-PC, SPARC, etc.) used in servers (10 million) and HPC systems (0.1 million).

There are several important developments taking place in these three market areas. The share of smartphones will cross the 50% level of the general phone market; the number of tablets sold in 2013 will be higher than the number of notebooks; and there has been a steady decline of desktop PCs sold.

The server market has been essentially constant over the last 6 years in terms of the number of units shipped and revenues (Figure 32). This is a sign of the saturation of the server market. The processing capacity for the HEP community is purchased from this server market, which as noted above is the smallest of the commodity chip productions, and practically constant in volume now.

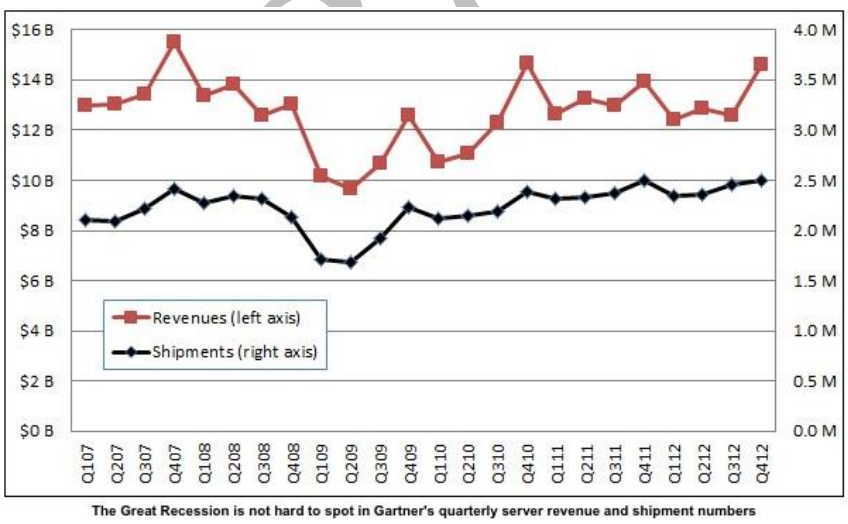


Figure 32: The (lack of) evolution of the server market in the last 5 years (courtesy Gartner)

The growth in chip complexity still follows Moore's Law with a doubling of the amount of transistors every 21 months or so[i].

The current generation of Intel processors is manufactured with structure sizes of 22nm and uses 3D transistor architectures. The competitors in this market (AMD and ARM) are about 1 year behind in terms of manufacturing processes using 28nm structure sizes and to some extent 3D transistors.

However, these considerations do not translate directly into corresponding cost improvements for complete servers. In that market the doubling of performance at a given cost has a period closer to 3 years.

There are currently three major trends to improve performance and total cost of ownership (TCO) in the server market:

1. Accelerator add-on boards based on slightly modified graphics cards (e.g. Nvidia, AMD), the new x86 based Xeon Phi (Intel) and some more rare designs using Xilinx FPGA architectures. The HPC community already makes considerable use of these architectures.
2. System-on-a-Chip (SoC) designs are the key for the manufacturing of low cost processors typically used in smartphones, tablets, and notebooks. The tight coupling of the CPU with the graphics unit including direct memory sharing allows for better energy saving measures and higher performance. These integrated designs are now also foreseen to enter the server market as micro-servers.
3. Micro-servers are a new server category based on lower-end processor designs (ARM, Atom) which couple sufficient performance with a much better performance/watt value. The focus so far is not on general applications but rather on specific ones (such as Facebook, Google, etc.). They could reach a server market share of about 10% by 2016.

All of these developments in server architectures could be potentially important and relevant to the HEP use of commodity architectures in future, and HEP should invest some effort in understanding how best to make use of them. However, today the price/performance of these new solutions is about a factor 1.5-2 worse than the current generation of servers we purchase.

It is not clear how these technology and market developments will affect the HEP community. The basic trend of price/performance improvements in the 25% range per year for 'standard' servers seems to continue into the future. Improving HEP processing efficiency by utilizing enhanced processor capabilities (e.g. vector units, GPUs, low end SoC) will require major code re-writes.

To answer some of these questions and to provide an environment where these new technologies can be tested and benchmarked, CERN is building a cluster of new architecture machines, supported by development and profiling tools. This environment will be used to help prepare HEP code for portability between traditional machines, and highly parallel architectures likely in the future.

4.1.1 Outlook for processors

- The technical roadmaps of the processor manufacturers are very challenging, but not unrealistic;

- There is a clear focus on the lower end consumer markets (phones, tablets), the server market, from which HEP buys its compute resources is at best stable;
- The costs of fabrication lines for new denser chips are becoming very significant, which only the largest companies can afford; possibly leading to close to single vendor markets in some areas;
- Price/performance gains come through the technology and fabrication of smaller structures, but due to the capital costs of such fabrication lines there may be a tendency to increase the cycle time between processor generations to maximise the investments. This could mean that the improvement of price/performance and power-consumption/performance slows, which in turn would imply increasing power costs;
- There is a general trend of increasing the numbers of cores on a chip which in turn has implications for I/O rates and thus performance needed on the disk systems.

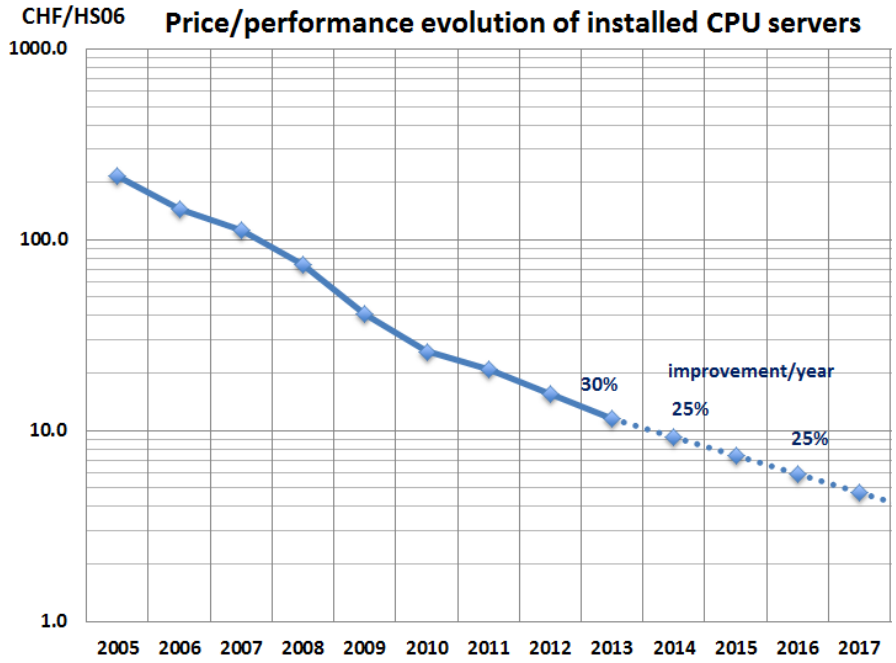


Figure 33: Evolution of price/performance of server systems, example of the CERN computer centre

Figure 33 shows, for the CERN computer centre, the recent trend, and the expected evolution for the next few years of the price/performance of CPU server systems. Continuing the present rates of improvement would lead to a 25% per year improvement in performance for a fixed cost over the next few years, assuming none of the perturbations noted above causes a change.

4.2 Disk storage

The two largest hard-disk manufacturing companies, Seagate and Western Digital, together have a market share of more than 85%. Revenues and shipments of disks have been declining over the last 3 years and this trend is expected to continue in 2013.

The floods in Thailand at the end of 2011 had a large effect on the industry, still reflected in the prices in Q3 of 2012. The main reasons for the problems of the disk vendors are based on the heavy decline of desktop PC sales, the move to solid-state disks (SSD) in notebooks and the extraordinary growth rates of smartphones and tablets. Revenues are expected to shrink by 12% in 2013, which corresponds to a shipment of about 550 million units (compared to 630 million units in 2011).

Figure 34 shows how storage technologies have evolved since 1980.

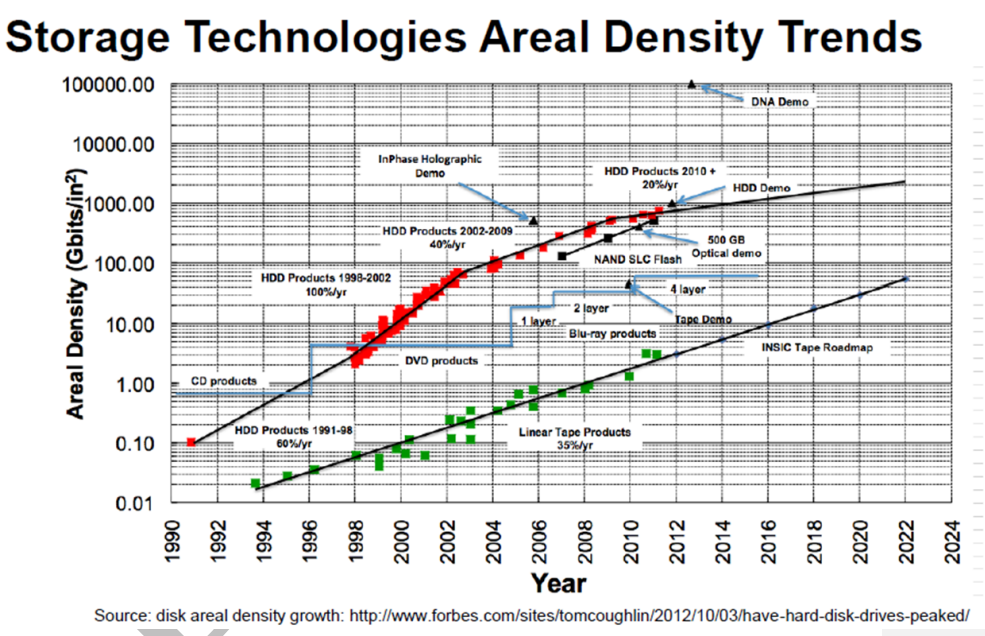


Figure 34: Storage technology areal density trends

Current hard disks use perpendicular recording, which essentially has now reached its limits at a density of 1Tbit/in^2. The next major technology jump will be to Heat Assisted Magnetic Recording (HAMR) and this is likely to lead to disk capacities of > 6TB some time in 2015. Two intermediate technologies, Shingled Magnetic Recording (SMR) and Helium filled disks, will start to take some market shares in 2014 with disk capacities of 5-6 TB, although the SMR technology will require operating system support and may not provide short-term benefit.

Figure 35 shows the likely evolution of disk drive densities and the expected introduction of new types of recording technology.

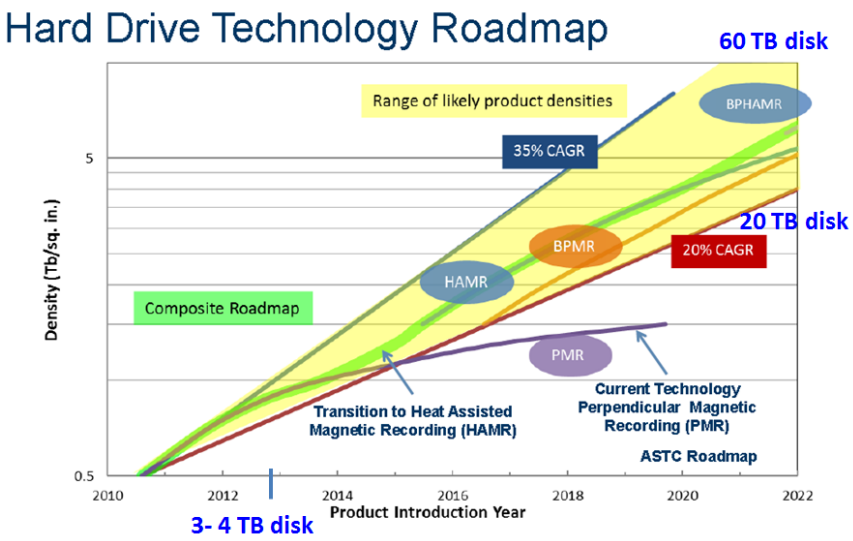


Figure 35: Disk technology roadmap

The market for solid-state disks shows large growth rates (of order 50%/year), but the volume of units shipped is still about one order of magnitude lower than for hard disks and the prices per unit storage are a factor of 5-10 higher.

There are two other points to be considered:

1. The large-scale move to smartphones and tablets in the PC market and the high investments needed for the next technology change will probably lead to a slowdown in the price/storage-volume improvements.
2. The capacity of hard disks has been continuously increasing while the sequential and random I/O performance has improved only very little. In addition, the performance advances in CPU servers are mainly due to an increase in the number of cores per CPU. Thus the ratio of I/O streams per disk spindle is increasing significantly and this could lead to higher levels of congestion in the disk storage layer.

Figure 36 shows the observed evolution of disk price/volume and its expected evolution for the next few years, using data from the CERN computer centre. An improvement of around 20% per year is anticipated from the technology, although some of the considerations noted above may have an impact on that.

In terms of prices the street price for disks is around 0.04-0.09 euro/GB depending on whether they are for the consumer or enterprise market. However, there is no particular observed difference in reliability in studies done at CERN between different categories of disk.

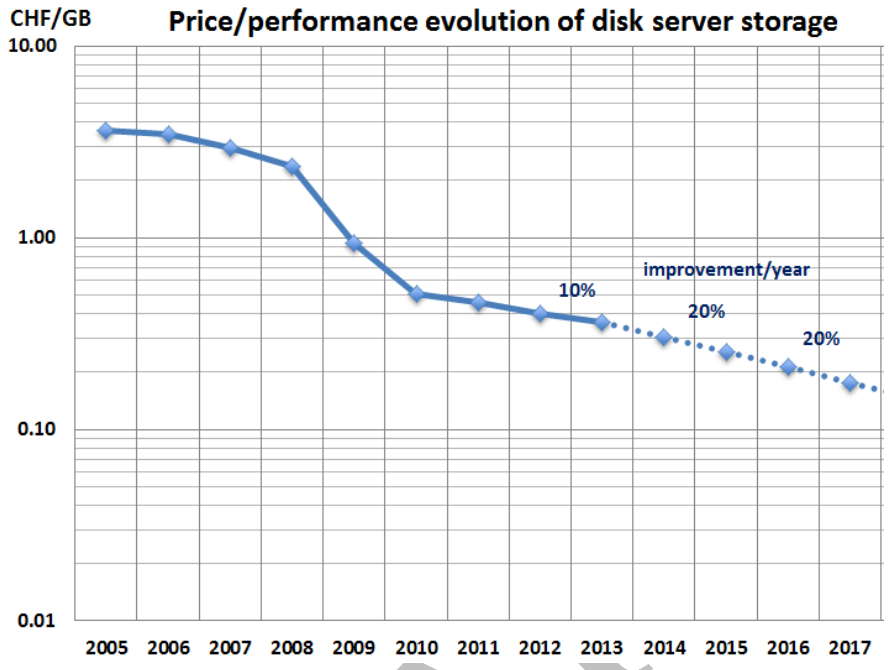


Figure 36: Evolution of price/performance of disk, example of CERN computer centre

4.3 Tape storage

The size of the market for tapes decreased by nearly 10% in 2012 following the trend of the last 4 years, that is expected to continue in 2013 (Figure 37). The market is dominated by the LTO format (93%), while the enterprise formats from IBM and Oracle make up around 2% of the volume. The technology still follows an exponential law with a density increase of about 32% per year. The current high-end cassettes have capacities of 2.5 TB (LTO), 4 TB (IBM) and 5 TB (Oracle).

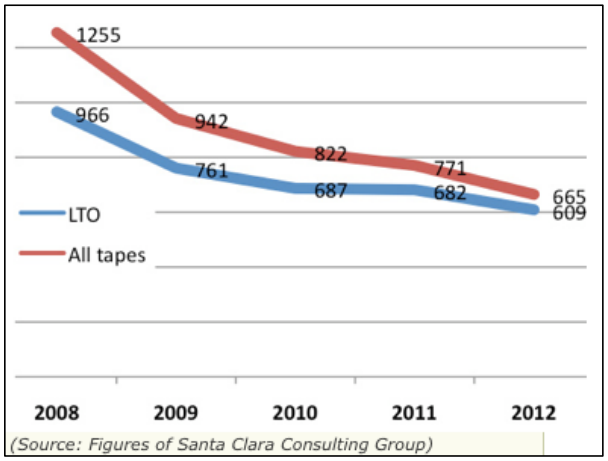


Figure 37: Tape cartridge market revenue ($million)

With the increase of cartridge capacity during the last few years we have seen a constant decrease of the cost per Gigabyte of storage (factor 2.5 over 3 years) today reaching a value of 0.03 -0.04 €/GB, which corresponds to a 30% improvement per year. The total cost improvement needs to include the installed tape robot capacity, the amount of tape drives and the yearly maintenance costs of the tape equipment. The effects of 'vendor-negotiation' is much higher than in the CPU and disk area. The overall fixed costs are about the same as the pure tape media costs, thus the effective improvement rate of tape storage is about 15% per year.

The replacement of tapes with hard disks for backup and archive storage is steadily progressing for small and medium size data sets. The TCO of tape installations for large amounts of data is still a significant factor better than those using hard disks. The 'death' of tape storage is still quite some years away. For the scale of LHC data, tape is still the most cost effective solution for archiving and long-term storage of data, from the point of view of cost of the storage as well as from the point of the low power consumption of tape storage compared to disk.

4.4 Networking

The enterprise networking market is growing rapidly and will reach revenues of about $43 B in 2013:

- The number of 1G, 10G, 40G and 100G network ports shipped on enterprise equipment in 2012 grew 22% over the previous year reaching some 360 million ports.
- The recent high growth rate for 10G port deployments in data centres has been accompanied by a steady price decrease of about 25% per year.

The estimated global data centre traffic is increasing, with an average annual growth rate of about 30%. It is interesting to make a comparison of this global

traffic with that of the HEP community. There we see differences of a factor 1000:

- Global traffic within data centres is around 2000 EB/year, while globally HEP traffic is ~2 EB/year;
- Global traffic between data centres is some 200 EB/year, with HEP traffic ~0.3 EB/year.

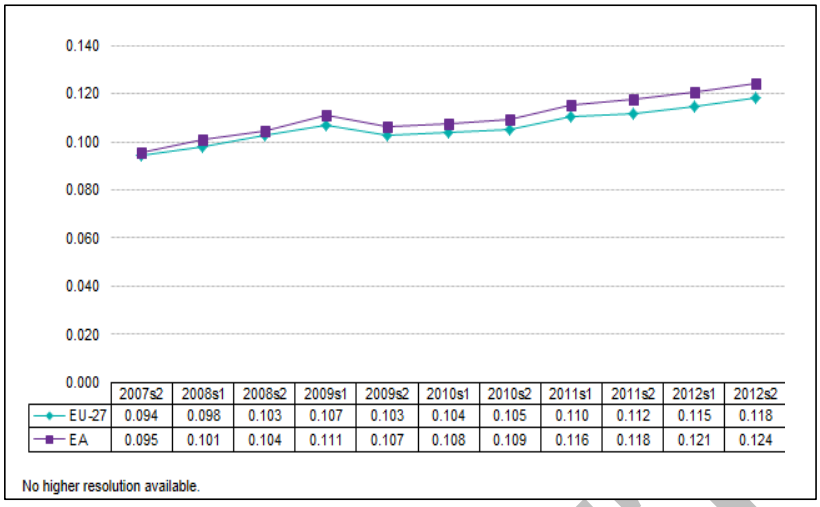
The 0.3 EB/year worldwide HEP network traffic can also to be compared with a volume of some 520 EB per year for the general global IP traffic, expected to be closer to 1000 EB by 2015-16. This growth is driven by the explosion in the numbers of connected devices, as well as rich content such as videos becoming more and more commonplace.

At the same time, the cost for commercially procuring 10G WAN Internet connectivity has decreased by about a factor of 3 in 3 years, and many academic backbone networks are already deploying 100G connections.

Thus, the likelihood is that the growth of HEP network traffic will not exceed the technical capabilities anticipated in the coming years, and it is reasonable to expect 10G connectivity to most large Tier 2 sites, and 100 G connections to Tier 1s and very large Tier 2s. There are of course some areas, the edges of Europe, parts of Asia, Latin America, and Africa where there will continue to be connectivity problems, and the HEP community should work together with the network providers to ensure adequate provision.

4.5 Overall Growth

The future growth of HEP computing resources will be limited by flat (or decreasing) budgets and the moderate price/performance improvement rates of computing equipment. Further increases in throughput have to come from optimizations in the computing models, using existing equipment with higher efficiency and following market trends, for example towards different processor technologies.



	2007s2	2008s1	2008s2	2009s1	2009s2	2010s1	2010s2	2011s1	2011s2	2012s1	2012s2
EU-27	0.094	0.098	0.103	0.107	0.103	0.104	0.105	0.110	0.112	0.115	0.118
EA	0.095	0.101	0.104	0.111	0.107	0.108	0.109	0.116	0.118	0.121	0.124

Figure 38: Evolution of electricity costs in Europe

It is important that efforts in the HEP community to improve efficiency are seen within a holistic view of achieving an overall improvement of TCO (or event throughput per unit cost). Changes in some areas might affect boundary conditions on a larger scale, including:

- Site spending profile (x% for CPU, disk, tape, LAN, WAN, services, operation, electricity): a dramatic improvement in CPU efficiency may require many more disks to maintain an adequate I/O rate, which may not improve the overall cost per event;
- Site computer centre architecture (e.g. flexibility to absorb new server formats, network blocking factors, rack and PDU layouts);
- Economies of scale for purchasing computing equipment (e.g. variations in technical requirements between experiments' requests);
- Timescale of developments versus the timescale of market and technology trends and changes (e.g. while a certain resource is optimized over long development periods, the topic becomes obsolete due to market shifts).

Until now the majority of Tier 2 sites (and many Tier 1s) have not had to pay the power costs directly. As the scale of computing grows, the electricity costs become significant, and many sites are moving to full-cost accounting, meaning that all of the costs including power must be budgeted for. Electricity costs continue to rise (see Figure 38), thus the fraction of budgets available for computing, storage, and network capacity improvements will tend to decrease. Rather than the 25% annual increase for compute capacity and 20% for storage from purely technological improvements, it is more realistic to assume 20% (CPU) and 15% (disk), to take these factors into account. Particularly as of a given budget, 75% can be used for capacity increases, and 25% is required for replacement of existing equipment (assuming a 4-year lifecycle).

It should be noted, however, that the extrapolations contain many assumptions with a lot of uncertainty and should only be regarded as reasonable guesses for the next 2-3 years. However, we will continue to update this forecast on a regular basis, following the technology trends.

5 Software Performance

5.1 Introduction

Major efforts have already been deployed to measure and improve existing HEP software in terms of improving CPU performance and reducing memory and storage requirements. However, if the future physics goals of the experiments at the LHC are to be achieved, further big improvements in computing performance will be required to process, calibrate, simulate and analyse the increasing data volumes that we expect to record.

In the past we have been able to rely on industry to deliver exponential increases in performance per unit cost over time, as described by Moore's Law. However the available performance will be much more difficult to exploit in the future since technology limitations, in particular regarding power consumption, have led to profound changes in the architecture of modern CPU chips. In the past software could run unchanged on successive processor generations and achieve performance gains that follow Moore's Law thanks to the regular increase in clock rate that continued until 2006. This has allowed software designs based on simple, sequential programming models to scale easily in terms of obtaining steady increases in performance.

The era of scaling such sequential applications is now over. Changes in CPU architecture imply significantly more software parallelism as well as exploitation of specialised floating point capabilities. Therefore it is timely to review the structure and performance of our data processing software to ensure that it can continue to be adapted and further developed in order to run efficiently on new hardware. This represents a major paradigm-shift in software design and implies large scale re-engineering of data structures and algorithms. The potential impact of this paradigm shift was presented at the European Strategy for Particle Physics Workshop, held in Krakow in September 2012 [1].

A dedicated R&D and upgrade programme for software and computing is therefore needed over the next decade in order to meet our future scientific goals at the LHC and to prepare for future experiments being planned by the world-wide HENP community. This programme should be seen as a continuation of the Multi-core R&D project, which was started by PH/SFT group working in collaboration with Openlab [2]. There is general recognition by the whole software community of the need to continue this work in order to address these issues. As a first step, an initiative has already been taken to create a new forum [3], open to the whole community, to make activities in this area visible. The goals are to explore, and in due course, establish a consensus on technology choices that need to be made, such as the best concurrent programming models, and a common view on new software components that should be developed for use in the data processing frameworks of the HEP experiments. An R&D programme of work has been started to build a number of 'demonstrators' for exercising different capabilities with clear deliverables and metrics. The goals are to identify the best tools, technologies (libraries) and models for use in HEP applications trying to cover all data processing domains: simulation, reconstruction and analysis. A regular series of bi-weekly meetings has been

running since January 2012 with a large and growing involvement by the whole community. A topical workshop was held in February 2013 at Fermilab to digest the results of on-going efforts. In the future, this may also lead to the establishment of a more formal mechanism for sharing knowledge and guiding future work.

In the following sections we start with a short summary of the physics requirements that set the goals on software performance and follow with an assessment of the impact of new CPU technology as a way of achieving these goals. We briefly summarise some of the on-going R&D work in various software domains in order to give an impression of the scale and complexity of the changes in software design and implementation that are implied. We then describe a proposal for creating a more integrated HEP-wide Software Collaboration that could help to facilitate contributions to the development of common software solutions requiring specialist knowledge and expertise. We conclude with a schedule for introducing planned major changes to software up until data-taking resumes after Long Shutdown 2, at which time the LHC will be operating at design energy and at full design luminosity.

5.2 Physics motivations

In the next 10 years, increases in instantaneous luminosity are expected that will largely exceed the available resources. With each progressive increase in the luminosity of the LHC the number of additional soft collisions per bunch crossing will increase, from the current level of 20-30 at $L = 7 \times 10^{33}/cm^2/s$ to up to 140 during operation of the HL-LHC [4]. This will result in a sharp increase in the overall per-event processing time and storage requirements.

In general we expect that the High Level Trigger (HLT) and event reconstruction will become relatively more costly with increasing pile-up events. For example, measurements made by CMS on modern Intel CPUs show that the HLT processing time increases more than linearly with increasing instantaneous luminosity [5]. The cost of simulation will scale to first order with the total number of events, whereas data analysis is a mix of activities, some of which scale as the number of events and some which scale as their complexity. Thus optimisation in all software domains can yield huge cost benefits in terms of overall computing resource requirements.

5.3 Impact of industrial technology trends

Recent years have seen a significant change in the evolution of processor design relative to the previous decades [6]. Previously one could expect to take a given code, and often the same binary executable, and run it with greater computational performance on newer generations of processors with roughly exponential gains over time, as described by Moore's Law. A combination of increased instruction level parallelism and, in particular, processor clock frequency increases ensured that expectations of such gains could be met in generation after generation of processors. Over the past 10 years, however, processors have begun to hit scaling limits, largely driven by overall power consumption.

The first large change in commercial processor products as a result of these facts

was the introduction of "multicore" CPUs, with more than one functional processor on a chip. At the same time clock frequencies ceased to increase with each processor generation and indeed were often reduced relative to the peak. The result of this was one could no longer expect that single, sequential applications would run faster on newer processors. However in the first approximation, the individual cores in the multicore CPUs appeared more or less like the single standalone processors used previously. Most large scientific applications (HPC/parallel or high throughput) run on clusters and the additional cores are often simply scheduled as if they were additional nodes in the cluster. This allows overall throughput to continue to scale even if that of a single application does not. It has several disadvantages, though, in that a number of things that would have been roughly constant over subsequent purchasing generations in a given cluster (with a more or less fixed number of rack slots, say) now grow with each generation of machines in the computer center. This includes the total memory required in each box, the number of open files and/or database connections, an increasing number of independent (and incoherent) I/O streams, the number of jobs handled by batch schedulers, etc. The specifics vary from application to application, but potential difficulties in continually scaling these system parameters puts some pressure on applications to make code changes in response, for example by introducing thread-level parallelism where it did not previously exist.

There is moreover a more general expectation that the limit of power consumption will lead to more profound changes in the future. In particular, the power hungry x86-64 "large" cores of today will likely be replaced, wholly or in part, by simpler and less power hungry "small" cores. These smaller cores effectively remove some of the complexity added, at the expense of increased power, in the period when industry was still making single core performance scale with Moore's Law. The result is expected to be ever-greater numbers of these smaller cores, perhaps with specialized functions, such as large vector units, and typically with smaller memory caches. Exploiting these devices fully will also push applications to make larger structural code changes to introduce significantly more fine-grained parallelism.

Although it is very hard to predict precisely where the market will end up in the long run, we already see several concrete examples that give indications as to the kinds of things that we will see:

- *Intel's Many Integrated Core (MIC) architecture*. This combines many smaller cores with very-wide SIMD units. The first commercial products (Xeon Phi) are in the form of a coprocessor and are aimed at the HPC market.
- *Systems implementing the forthcoming ARMv8 64bit architecture*. Here the significant use of the ARM processor in low-power or embedded systems (e.g. mobile devices) positions it well to enter a server market dominated by the power limitations described above, even if the route it followed to get there differs from that of Intel. Intel is also preparing its own low power server variants, hopefully leading to a competitive market with price benefits for buyers.
- *General Purpose Graphics Processing Unit (GPGPU or GPU),* such as the Tesla accelerators from NVIDIA.
- *AMD Accelerated Processing Unit (APU)* combining GPU, CPU and specialised

processor technologies on the same chip.

Overall the market is likely to see significantly more heterogeneity in products than in the past couple of decades. Effectively exploiting these newer architectures will require changes in the software to exhibit significantly more parallelism at all levels, much improved locality of access to data in memory and attention to maximize floating point performance through enhanced use of vectorisation, instruction parallelism and pipelining. Most of the scientific software and algorithms in use today in LHC experiments was designed for the sequential processor model in use for many decades and require significant re-engineering to meet these requirements.

If we fail to meet the challenge of adapting the software, the cost of computing required for the luminosity upgrades of the LHC will not profit from Moore's Law cost reductions as in the past. Market trend studies made by CERN [7], for example, indicate that expectations of overall throughput/cost gains should decrease from 40% per year to 20% per year for typical low-end servers with multicore CPUs that we use for high throughput computing. This corresponds to the doubling time" for performance/cost roughly increasing from 1.5 years to 3 years. Only by embracing the newer architectures are we likely to have sufficient computing power for our scientific goals over the next ten years. It is also important to recognize that if we obtain a full Moore's Law-like gain by doubling of performance every 1.5 years, we are talking about 2 orders of magnitude over the next 10 years and 3-4 orders of magnitude through the HL-LHC era. Achieving these huge potential increases will likely transform completely the processor landscape and software design. Investigations to upgrade the software to the near/medium term processor products should be seen as steps along an R&D path in the longer term eventually aimed at efficient scalability of our applications through order of magnitude increases in processor power.

Traditionally HEP experiments have exploited multiple cores by having each core process in parallel different HEP 'events'; this is an example of a so-called *embarrassingly parallel* problem that results in speedup factors that scale with the number of cores. However, as already mentioned, a trend towards many (100's) cores on a single socket is expected in the near future, whilst technical limitations on connecting them to shared memory could reduce the amount of memory and access speed seen by a single core. This causes a major problem for experiments running at the LHC, since planned increases in luminosity are expected to result in more complex events with higher memory requirements. One is led to conclude that in future we will need to efficiently use multiple cores to process a single event i.e. move towards finer-grain parallelism in our data processing applications. This needs to be done both to improve throughput reducing memory requirements (footprint and bandwidth) and to expose parallelism to the algorithms that can be then optimised on advanced CPUs (incorporating vectorisation, instruction pipelining and so on), or on GPUs.

5.4 Areas of Research and Development

In this section, we describe what we believe are the relevant elements of an R&D and upgrade program that are necessary to meet the challenges posed by the new heterogeneous processor environment. A program to ensure that our

software will be sufficiently scalable and efficient on these architectures can be seen as an upgrade effort through the end of this decade, accompanying the phase-1 LHC detector upgrades, and an R&D effort towards the eventual software we will need for the phase-2 upgrades and HL-LHC. More generally this R&D activity will also benefit the planning of future HENP experiments, such as FAIR, ILC/CLIC and TLEP, as well as helping to guide future hardware purchases.

Our premise is that HEP software will need to accommodate the new hardware architectures by introducing parallelism whenever possible. This is in order to make efficient use of all the available cores and to exploit micro-parallelism inside the CPUs. As other resources will not scale with the number of cores, it implies reducing requirements for IO, memory etc., on a per core basis. In some applications, in particular triggering, an important requirement is reducing latency, which can be achieved by making use of all available cores to run a job in less time. Beyond this, programs will have to be optimized to make use of lower level parallelism offered by the CPU micro-architectures. Efficient exploitation of vector registers (SIMD), instruction pipelining, multiple instructions per cycle and hyper threading will be key to reap the benefits of the new architectures in order to follow Moore's law for the coming years. It will also be essential to evaluate what the role of GPGPUs will be as well as of the new devices appearing on the market, such as the Intel Xeon Phi as well as low power CPUs, such as ARM or Intel's Atom CPU. All these explorations are made more challenging by the rapidly evolving technology and the lack of standards for parallelisation and vectorisation in computing languages.

One of the main tasks will be to study and evaluate the various hardware solutions that are currently available in the context of each of the various computational problems we have to deal with. Each parallel hardware option comes with its own constraints and limitations (e.g. memory organization and available bandwidth, in the required programming language or programing model etc.) and these all have to be understood for taking the best decisions and ensuring scalability in the longer term. For example, we may find that the use of GPGPUs is very effective in some of the algorithms used in the reconstruction or simulation, whereas in others the use of such processors is not cost effective due to large data input and output requirements. New architectures such as the Intel MIC could also be useful to offload large computations to accelerators comprising many cores, whilst maintaining the flexibility of reusing the bulk of the code written in C++. For this reason it is thought that a period of technology evaluation is now needed, developing prototypes ('demonstrators') to exercise the different options that we have at hand.

5.4.1 Concurrent Programming Models and Software Frameworks

Concrete algorithms can be run in parallel by making use of threads, but integrating them to run as required in a single application is highly non-trivial. The ability to schedule algorithms to run concurrently depends on the availability of the input data each algorithm requires in order to function correctly. An analysis of the data dependencies in existing applications shows that the potential 'concurrency factor' is rather low. Moreover, in current applications there are always specific algorithms that take a large fraction of the total processing time (e.g. in the tracking code) and this can be a strong limiting

factor on the scalability of the overall performance when adapting to run on many cores. Options for addressing this include making the algorithm fully re-entrant (not easy), instantiating several copies of the algorithm each running on a different event concurrently, or parallelizing the algorithm. All imply a major redesign and rewrite of the algorithmic code. A new approach to processing many events in parallel will also be needed in order to avoid the long tails in the processing time distribution of many events.

We conclude that parallelism needs to be added at all levels at the same time, the event level, the algorithm level, and the sub-algorithm level. Software components at all levels in the software stack will need to inter-operate and therefore the goal should be to standardize as much as possible on basic design patterns and on the choice of concurrency model. This will also help to ensure efficient and balanced use of resources.

Running experiments have a significant investment in the frameworks of their existing data processing applications (HLT, Reconstruction, Calibration, Simulation and Analysis) and it is realistic to expect that collaborations will favour incorporating new vectorization and parallelization techniques in an adiabatic manner, whilst still preserving the quality of the physics output. The introduction of these new techniques into the existing code base might have a rather confined local impact, in some cases, as well as a more global impact in others. For example the vectorization of some inner-loop calculation (' hot spot ') may affect only a very concrete algorithm and could be done independently of many other optimizations that can be applied to the code. On the contrary, being able to run on many concurrent events with the aim of reducing the memory footprint in a multi-threaded application may require rather deep changes in the framework and services used to build the complete application and therefore will have much larger consequences. The goal should therefore be to develop a new set of deliverables, in terms of models and framework services, that can be integrated into the various existing data processing frameworks and which can assist physicists in developing their software algorithms and applications.

As an example, the Gaudi framework [8] is being redesigned to embrace concurrency. This modified Gaudi framework will be used by ATLAS and LHCb to test and evaluate the impact of adding concurrency to their data processing applications. For this a prototype called GaudiHive [9] has been created, which follows the concept of task parallelism. Here a task executes a given *Algorithm* on a given *event*. The dependencies of these tasks on each other can be expressed in terms of a directed acyclic graph (DAG) formed from the input-output relation of the *Algorithms*. A slice of the reconstruction application of LHCb has been successfully adapted to this new concurrent framework delivering the same physics results as the sequential version. This has been achieved by preserving intact the algorithmic code written by physicists. The preliminary results on memory usage and scalability are very promising.

CMS is also working on evolving changes to its core software framework (CMSSW) and the related code infrastructure that provides the processing and event data model. An implementation of the CMS Framework allowing for parallel (threaded) execution of the existing Framework modules on multiple x86-64 cores is being prepared for use for the LHC run starting in 2015. In the

following years, this will evolve significantly as the algorithms and data structures are re-engineered to bring out the more fine grained parallelism required for scalable and efficient use of the new processors.

Depending on the results of the investigations mentioned above, additional technologies such as OpenCL, CUDA, and/or others still to appear, will need to be introduced and interfaced with the full processing framework. Given that on the time scale of HL-HLC it is likely that several orders of magnitude increase in parallelism may be required, the tools available today will surely not be the last word.

5.4.2 Event Simulation

One of the most demanding applications is that of detector simulation. The precision of the simulation algorithms, such as those contained in Geant4 [10], has reached a very high level, allowing a detailed reproduction of the experimental results and an unprecedented understanding of the detector. This precision is however attained at the price of rather demanding algorithms. This fact, compounded with the intrinsic "slow" stochastic convergence of Monte Carlo calculations, which is proportional to the inverse of the square root of the number of events, explains the high computational demands for simulation.

Simulation with Geant4 represents a very important fraction of the total CPU used by LHC experiments, as well as by many other HEP experiments. It has, and continues to be, the focus of optimization activities by a number of groups. As a toolkit used by many in the HEP community, supported by a worldwide collaboration, there are some existing efforts in a number of places to understand how to evolve Geant4. In addition, the experiments have also developed "fast simulation" frameworks that, at the price of different levels of simplification, offer substantial speedups. These "fast simulations" are validated both with the data and with the results of the "full simulation".

The currently ongoing efforts by the Geant4 collaboration to integrate changes from a thread-parallel prototype [11] are an important first step and will undoubtedly inform other efforts. We note in addition another important potential gain from a scalable, multithreaded geant4 simulation, which eventually supports a heterogeneous range of hardware. As simulations based on Geant4 typically have rather modest input data requirements, relative to data reconstruction for example, and significant CPU use, they are perfect candidates for exploiting "opportunistic" resources or even "volunteer computing". In opportunistic computing, we use computing clusters owned by others and often designed for other types of workflows. In volunteer computing, private individuals donate spare computing cycles, typically on their desktop machines. In both cases, the ability to use whatever processor hardware is available, while simultaneously limiting as much as possible memory use and the amount of output data held resident on the host machine, will maximize the potentially usable resources.

Taking all this into consideration, it is clear that simulation is one of the primary targets for optimization on the new computer architectures. In contrast to reconstruction and analysis, a large fraction of the simulation code is experiment independent and therefore any improvement in its performance will

immediately benefit all experiments. This matter has been studied for more than one year by a small team, with the objective of developing a simulation framework prototype [12] that could allow optimization of both full and fast simulation at all possible levels, from multi-node to microscopic optimization at the level of the CPU microarchitecture and of GPUs. One particular area of study has been to explore very different ideas in the development of a radically new 'particle transport' code. The prototype is still very preliminary, but it shows promising results as far as navigation in complex geometries is concerned. The next steps will be to introduce some realistic physics processes and simulate the scoring inside the detector. When a convincing architecture is defined, the physics processes from Geant4 will be ported into the new framework, applying the coding and algorithmic changes required for optimization on parallel architectures. With more realistic physics models and with the further optimisation of the communication between the threads, it is hoped to achieve big gains in performance.

An activity conducted in close collaboration with this prototyping work is the exploration, performed by the HEP-ASCR[4] team at Fermilab, of the use of CPU/GPU hybrid systems. This particle based vectorization approach aims to eventually parallelize Geant4 at the level of secondary particles or "tracks" generated as the incident primary particles traverse the material and electromagnetic fields in the detectors. Members of the Fermilab and CERN teams meet bi-weekly to share progress on the prototype, and with the ASCR team to discuss issues specifically related to computing performance.

5.4.3 Event Reconstruction

Many of the important algorithms used in the HLT and the offline event reconstruction, as well as some aspects of the data analysis, are such that their cost in CPU time increases non-linearly with luminosity and in particular with the combinatory effects resulting from increases in the number of pile-up events. For this reason, we expect their relative importance to the overall cost to increase, and thus we expect that significant effort will be necessary here. Eventually many portions of the code will need development to achieve the necessary scalability and efficiency. For example, a concrete and important area that will need to be addressed is the Tracking.

There are several possible approaches for parallelizing the track reconstruction. The simplest solution is to parallelize the seeding, building, and fitting steps individually. For the seeding, the detector can be divided into regions while for the track building (fitting), the input seeds (tracks) can be divided into groups. One downside to this approach is the necessary synchronization after seeding and building to check for duplicate tracks. This approach is minimally invasive and provides the necessary scaling for additional full function cores. However, each process will still need to perform a variety of complicated instructions, limiting the applicability of this approach to systems with multiple simple cores.

There are alternative track reconstruction algorithms which are more inherently parallel and which can take advantage of many simple cores and vector

[4] US-DOE Advanced Scientific Computing Research (ASCR) program

instructions. Two examples of such approaches are the Hough transform and cellular automata. The Hough transform works by applying a conformal transformation to all hits in the detector such that all hits belonging to the same track cluster in a well defined way. Track finding amounts to locating clusters of points rather than iteratively traversing detector layers. The conformal transformation and the cluster finding should be good candidates for vectorization and parallelization with simple cores. One possible implementation of a cellular automata approach is to find all combinations of three hits that are consistent with a track and then combine the triplets together. The triplet finder can be made simple, allowing it to be vectorizable, especially when effects of material are ignored.

The track reconstruction performed during offline processing of the data is similar to that performed by the high level trigger (HLT) and the current hardware is also similar. Thus, most changes to the track reconstruction will provide benefits to both regimes. However, at the trigger level the timing concerns are much more critical. Therefore, in the future, the hardware used by the HLT (or earlier trigger) may become more specialized and be the first to take advantage of the new architectures. Thus, the trigger may become the logical testbed for new implementations of the track reconstruction.

5.4.4 Input and Output of data

Although we focus primarily on processor technology in this paper, we note that I/O concerns are also relevant here in two different ways. Firstly, simply feeding sufficient input data to the individual processor units and insuring that outputs created are collected from them in an efficient, scalable way is likely to be a major challenge as the required parallelism increases by orders of magnitude. For example, today's sequential applications are typically designed to expect explicit serialization points for outputs, which will likely cause scalability problems as we move to highly concurrent, parallel applications.

Secondly, in order to use vector units efficiently and to reduce stalls resulting from accessing data not in the processor memory caches, there will be more emphasis on data structure layout and data locality than in the (object oriented) past. Simpler data structures such as structures of arrays, as opposed to arrays of structures, will likely become more common and software algorithms will need be adapted.

5.4.5 Development Tools and Libraries

We will need increasing agility in porting HEP software to a variety of new platforms. There will surely be significant evolution in compilers and associated tools, and perhaps also on operating system support. In addition some number of auxiliary support tools for performance profiling, code analysis and debugging will be required. Given the more complex nature of parallel programs and the sorts of specialized architectures we may see, code profiling will be an important activity to ensure efficient use of the processors and find limits to scalability. General-purpose tools like IgProf [13], as well as more advanced and specialized profilers, will likely be needed more widely. Similarly run-time debugging applications which will be much more complex than today's simple sequential applications will need more advanced tools, which simplify the process will be

critical. In order to ensure code quality and find potential faults early for newly written code, we also expect that code analysis tools will become increasingly important to our code development and integration process.

One important class of support tools is math libraries. During the frequency scaling era for CPU's and in particular with the transition to object oriented programming, general interest in the numerical aspects of programming arguably waned somewhat. This is changing. The combination of the transition to x86-64 (which brought the transition from x87 to SSE2 floating point), more dynamic open source compiler development (gcc since version 4, LLVM) and the need to use the new architectures, has renewed interest in numerical computing, vectorization, etc. One example of this is the VDT [14] library, which provides inlineable, vectorizable versions of mathematical functions and allows tradeoffs between accuracy and speed. Experiments will need to investigate the use of this and other such tools to understand when and where they can be used.

Finally, the software itself should be able to manage heterogeneity at the level of an individual worker node. However, both during the initial introduction of new architectures and in the long run, it will still be necessary to manage the heterogeneity between clusters of resources on the grid. Support for such heterogeneity will be needed both at the level of the workflow management tools and in the grid software.

5.4.6 Addressing Software Maintenance Issues

It is well known that projects with very long lifetimes pose significant challenges for maintenance of the related software systems. The lifespan of LHC software is more than 30 years and in that time it is subjected to continuous change. Without constant attention software rapidly accumulates attributes that inevitably result in a significant degradation in reliability and performance. Although the operational phase of the LHC started relatively recently, the development of LHC software started more than 15 years ago. Our software packages have evolved greatly over time, growing more complex in the process. A period of refactoring is now required in order to ensure its future maintainability. This involves

- adapting to new requirements, in particular those relating to future operation of the LHC
- evolving to use more powerful software standards and technologies,
- removal of obsolete functionality,
- the further identification and removal of software defects so as to prevent system failures.

For example, development of LHC software started when OOP and C++ were in their infancy. Since that time, standard libraries have evolved to include functionality that was originally developed in-house, such as that provided by the standard C++ library and by Boost. It is important to following the evolution of these 'external' packages such that our software is adapted to use them, whenever there is an advantage to be gained from doing so. Programming languages, such as C++ and python, continue to evolve and it is important to adhere to new versions of the coding standards. For example, considerable effort is currently being invested to re-develop the ROOT interpreter, using

LLVM/clang technology, so that it can easily be adapted to conform to the C++11 standard and future versions of the standard when they appear. The permanent upgrade to new software technology is a pre-requisite to keep the software future-proof and agile enough to cope with the rapid change of hardware technology. Virtualisation technology, such as that provided by the CernVM toolset, is helping to facilitate the transition to new computing platforms.

Given the increasing processing demands, as well as the complexity that results from software parallelism, it is important to invest in improving the quality of existing code base in order to minimise the overall effort that will be required to maintain it. Removing code that implements functionality that is never used helps to simplify maintenance. One example of this is the current work being done to streamline 'physics lists' of Geant4. Implementing thread safety in core libraries was not considered essential until the recent work on fine grain parallelism started. Campaigns have started to improve the thread safety of core libraries and to document those parts that are not thread safe.

A large fraction of the effort available is needed to cover these typical maintenance activities. Experience shows that over a period of 10 years there is a large turnover in manpower, so preserving knowledge over time is a real concern. An important element in addressing the associated risks is the provision of up-to-date documentation. Achieving a better common understanding of each other's requirements and the software that exists can lead to setting up common projects and a more efficient use of the effort and expertise that is available across the community. Single efforts in that direction have already started, such as a combination of the ATLAS and CMS workflow management tools, as well as the consolidation of the physics validation software used by all four experiments.

5.4.7 Training in development of concurrent software

As the development of efficient and thread-safe code is even more challenging than the current development activities, a systematic knowledge exchange is essential. Such share of knowledge happens very successfully via the concurrency forum. However, the discussions there already require a very high expertise in the use of various software development techniques.

To enable more members of the community to actively participate in the various parallelisation projects, dedicated training efforts are needed. First attempts have been regular courses organised by the CERN OpenLab project and the first thematic CERN School of Computing on "Mastering State of the Art Computing" [15] this summer. In addition, the ESC series of Schools organised by INFN have given very valuable training opportunities for researchers working on improving the performance of scientific computing applications [16]. There are on-going discussions to make such training available to an even larger audience and to provide a regular parallel software design course shared by the LHC experiments.

5.5 A HEP Software Collaboration Initiative

The programme of work described above is both essential and ambitious. While the goals are clear, the way to reach them are still the subject of wide-ranging

R&D, made more uncertain, and challenging, by the shifting technology landscape. A necessary condition for the success of such an endeavor is to establish close collaborations with those who are solving similar problems in research and in industry and to optimize the usage of resources within HENP. This will not be possible without building a strong collaborative effort within the HEP community. The creation of the "Concurrency Forum" [2], was a first important outcome of the "Workshop on Concurrency in the many-Cores Era" [3] held at Fermilab in November 2011 to explore the possibility that interested HEP institutions and projects collaborate on R&D related to concurrent frameworks and applications. Fermilab also hosted the first annual meeting of the forum in February 2013 with the participation of institutions involved in both the LHC program at CERN and the ASCR program at Fermilab. The forum has been meeting bi-weekly via videoconference for ~2 years and offers individual researchers the opportunity to share progress with the world wide HEP software and computing community. One of the important benefits of the Concurrency Forum is the sharing of knowledge within the HEP community, achieved through building demonstrators to show various capabilities and by having a forum within which to disseminate that knowledge.

Building on the successful experience of the Concurrency Forum, there is a recognised need for a somewhat more formal framework within which to continue and expand this work. A proposal to expand the Concurrency Forum into a more formal HEP Software Collaboration is under discussion, in order to provide a framework for increasing the level of collaboration and involvement amongst colleagues with experience in concurrent programming. Moreover the aspects of Grid and Cloud computing, which are not covered by the Concurrency Forum, should be brought into the global picture of the evolution of the HEP computing model. We must recognise that there is expertise in many of our HEP institutes and in other scientific communities, and we should try and make use of that expertise. Having a more formal collaboration would bring the means to provide recognition and give credit to contributors to the effort. The experience from the LCG project shows that giving recognition, and credit, is important in bringing potential collaborators into a common effort. It is important, however, that this proposed collaboration not be limited to LHC experiments but should be HEP-wide.

It is also important to provide roadmaps and priorities for investigation and development. Having such roadmaps would allow potential collaborators to see in which areas they can contribute or where their expertise may be relevant.

The collaboration would build on the work of the Concurrency Forum in building expertise in the HEP community at large. We need to attract new people with the required skills and to transfer that knowledge within the community.

Ultimately, it may become important to consider a team of experts who are able to provide consultancy and practical help in helping experiments to optimise their code. This collaboration could be a starting point for building such an optimisation task force or service. Of course, having such a team would only be realistic if a lot of the software services of the experiments had common foundations (libraries, frameworks, tools), and one potential important outcome of this proposed collaboration would be to build those common foundations.

5.6 Timeline for re-engineering LHC software

Important milestones are defined by the LHC upgrade plans and the schedule for long shutdowns (LS), as these provide windows of opportunity for making major changes that require extensive validation before they can be used in production.

Concerning core software, the immediate plans are focused on the release of major new versions of Geant4 and ROOT. Version 10 of Geant4 will have support for multi-threading, whilst Version 6 of ROOT has a new reflection system and interpreter based on LLVM/clang technology. The schedule for these releases was chosen to give the experiments sufficient time to integrate them in their data processing applications and to test them extensively during LS1. LS1 will also be an opportunity for further campaigns to improve performance of the existing experiment production software in order to meet the demands of data-taking in 2015, when the LHC will be operated at design energy and nominal luminosity.

The work that needs to be done to introduce concurrency has already started and will ramp up during LS1 at a rate permitted by the available effort. CMS has set the ambitious goal of releasing a multi-threaded version of CMSSW, the CMS data processing framework, for use in production in 2015. More generally, the aim should be to deploy applications supporting concurrency during LS2 (2018), so that advantage can be taken of the shutdown period in order to test extensively on the computing infrastructure that will be deployed in the computing centres and to validate the consistency of physics results. By this time it is also intended to deploy a new simulation toolkit with a new particle transport 'engine' that optimises use of memory caches and that exploits vectorization capabilities of the CPU. Work will also continue on improving the performance of the Input/Output (IO) provided by ROOT by further exploitation of parallel techniques. This work is required in order to meet the demands of running after LS2 at the full design luminosity of the LHC.

5.6.1 References

[1] F.Carminati, J.Harvey, P.Mato Addressing the challenges posed to HEP software due to the emergence of new CPU architectures, CERN PH/SFT Input to the European Strategy for Particle Physics Update, Krakow, September 2012

[2] Multi-Core R&D, https://twiki.cern.ch/twiki/bin/view/LCG/MultiCoreRD

[3] The Concurrency Forum : http://concurrency.web.cern.ch

[4] L Rossi and O Brning. High Luminosity Large Hadron Collider - A description for the European Strategy Preparatory Group. Technical Report CERN-ATS-2012-236, CERN, Geneva, Aug 2012.

[5] P.Elmer, S. Rappoccio, K.Stenson,, P.Wittich, The need for an R&D and Upgrade Program for CMS Software and Computing, June 2013

[6] Samuel H. Fuller and Editors; Committee on Sustaining Growth in Computing Performance; National Research Council Lynette I. Millett. The Future of Computing Performance: Game Over or Next Level? The National Academies Press, 2011.

[7] http://wlcg.web.cern.ch/news/technology-market-cost-trends

[8] http://proj-gaudi.web.cern.ch/proj-gaudi/

[9] B. Hegner at al, Evolving LHC Data Processing Frameworks for Efficient Exploitation of New CPU Architectures, Proceedings IEEE-NSS 2012

[10] http://geant4.cern.ch

[11] Xin Dong, Gene Cooperman, and John Apostolakis. Multithreaded Geant4: Semi-automatic Transformation into Scalable Thread-Parallel Software. volume 6272 of Lecture Notes in Computer Science, pages 287{303. 2010.

[12] J. Apostolakis, R. Brun, F. Carminati, Andrei Gheata Rethinking particle transport in the many-core era towards GEANT 5, 2012 *J. Phys.: Conf. Ser.* **396**

[13] http://igprof.org18

[14] https://svnweb.cern.ch/trac/vdt

[15] The first thematic CERN School of Computing: https://csc.web.cern.ch/CSC/2013-tCSC/This_year_school/This_year_school-t2013.htm

[16] ESC International Schools :

http://web2.infn.it/esc09/

http://web2.infn.it/esc10/

http://web2.infn.it/esc11/

http://web2.infn.it/esc12/

6 Experiment Software Performance

6.1 ALICE

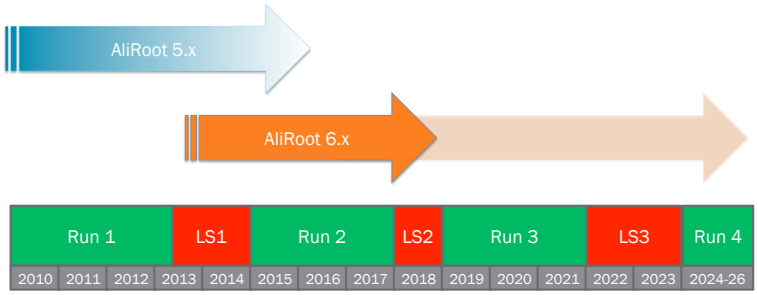


Figure 39: Timeline for developments of ALICE software framework

The current ALICE software framework (AliRoot 5.x) is almost 15 years old and was used to fulfill all the use cases for data processing in ALICE starting from calibration and reconstruction to simulation, analysis and visualization. The framework is built on top of ROOT and directly depends on it. On the other hand, it seamlessly interfaces to AliEn and PROOF allowing users to extend their analysis to Grid for batch processing or to one of several dedicated analysis facilities running PROOF (such as CAF at CERN) for interactive work.

While the current framework suffers from several problems (the most notable being considerable memory consumption often breaking the 2 GB/core limit) which is no surprise given its age, it still works sufficiently well to let everyone do their work. The existing event data model and user interfaces will be difficult to change without disruptive effects. This is why we plan to start the next iteration of AliRoot (6.x) during LS1 and that version will be entirely independent of the current code.

In this new framework we will simplify the data model in order to improve I/O performance and we will write the code in such a way that it can potentially benefit from special hardware such as GPUs or coprocessors like Intel MIC. In this respect, we will also revisit and try to improve the existing algorithms for cluster and track finding and fitting. If any of these developments results in substantial improvements that can be retrofitted into AliRoot 5.x, we will try to do it but the priority will be to consolidate AliRoot 5.x and keep it as stable as possible while doing all developments in the AliRoot 6.x branch.

In the following chapters we describe the most important use cases that are covered by the AliRoot framework along with possible improvements for Run2.

6.1.1 Calibration

In its present form the different steps of the ALICE detector calibration are carried out in both Online and Offline environments.

6.1.1.1 Online calibration

The ALICE online calibration is based on the data coming from the three online systems DAQ, DCS, and HLT. Here, dedicated procedures called Detector Algorithms (DA) are run on the raw data collected by the experiment. A common framework named Shuttle is in charge of processing the DA output aimed at producing the calibration objects that are stored as the calibration data into the ALICE Offline Condition Database (OCDB) on the Grid. The Shuttle processing (both globally and in terms of preprocessors) is monitored through MonaLISA and published in the ALICE Electronic Logbook.

6.1.1.2 Offline calibration

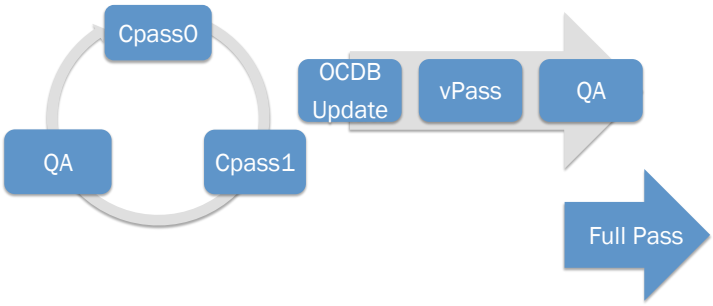


Figure 40: Offline calibration steps and procedures

Some of the ALICE calibrations rely on information that comes from reconstruction. The offline calibration procedure is divided into two stages, commonly referred to as CPass0 and CPass1 (where CPass stands for Calibration Pass), in cascade one to the other, and followed by a Quality Assurance (QA) process in case of CPass1.

Each CPass consists of two separate jobs run on the Grid. The first one is a chain of reconstruction and calibration algorithms that collect the information that will be used to produce the calibration parameters.

The second step of CPass is a merging job, that, looping over the calibration files produced for a given run, elaborates the final calibration data and stores them in the OCDB.

As soon as these are available on the grid, and every detector confirms its readiness both in terms of QA and calibration, a Validation Pass (VPass) is started. This consists of the same reconstruction and QA that is run for the final results; with the only exception that only a limited amount of the statistics is used (~10%). If the outcome of the quality checks of the VPass data is positive for all detectors, then the final Production Pass (PPass) is started.

A major drawback of the current ALICE calibration schema is the time spent in the offline passes. This can be reduced by moving as many calibration procedures as possible to the Online domain.

The first step towards a more efficient calibration model is the removal of CPass0. This can be done only provided that the quality of a single reconstruction pass aimed at calibration is good enough to allow the various detectors to produce optimal calibration data. The basic prerequisite is that the main ALICE tracking detector (i.e. the TPC) is capable to perform its calibration online. The plans for Run2 include a first implementation of such procedures, in a joint effort between HLT and TPC, which could eventually also involve other detectors.

6.1.2 Simulation

GEANT 3 is the transport Monte Carlo code used extensively by the ALICE community for simulation of the detector response. However, it is no longer maintained and has several known drawbacks, both in the description of physics processes, particularly hadronic, and of the geometry. ALICE has spent considerable effort in evaluating GEANT4. The virtual Monte Carlo interface allows us to run full ALICE simulations also with GEANT 4 and to compare them with the GEANT 3 results; the advantage being that both simulation programs use the same geometry and the same scoring routines.

This exercise has now concluded that the output of two simulation programs results in compatible results without impact on physics performance. This opens the possibility for use of GEANT 4 for ALICE simulation during Run2. However, the CPU performance of GEANT 4 transport, as it is currently implemented in AliRoot using the virtual Monte Carlo interface, is still worse by a factor of 3 compared to GEANT3. This poses a significant problem for the adoption of GEANT4 and we are committed to work closely with GEANT4 and other experts from CERN-SFT group to resolve these differences and improve the speed of GEANT4 based simulation for ALICE. First results already show that we could potentially gain up to 15% in speed by doing a small change in how memory allocation is managed within GEANT 4. After preliminary tests we estimate an additional 18% gain in performance can be achieved by using modern vectorized math libraries.

In the long run we also expect to profit from on-going developments of GEANT 4 in terms of adapting it to run more efficiently in multi-core environments and possibly utilize resources such as GPUs. This becomes particularly important as there are emerging common ideas in ALICE and ATLAS and concrete proposals to utilize spare computing capacity on some of the world's largest supercomputers in an entirely opportunistic way. While we can in principle run our software in its present form in this environment, we do not really benefit from the existing GPU capacity and parallel supercomputer architecture and this is the area where we expect GEANT 4 to develop and fill the gap.

While we count on all these developments it is clear that they will not be able to fully compensate for increasing CPU needs. In order to improve the situation we will have to resort to alternative strategies for simulation such as parameterizations and embedding.

To reach the required sample size, fast simulation methods based on meaningful parameterizations of the results from detailed and consequently slow simulations must be applied. The systematic error introduced by the

parameterizations is in general small compared with the reduction of the statistical error.

The current AliRoot framework provides base classes that allow for a representation of the detector or detector systems as a set of parameterizations of acceptance, efficiency, and resolution. The Muon Spectrometer fast simulation has already been implemented using these classes.

The simulation of small cross-section observables would demand prohibitively long runs to simulate a number of events commensurate with the expected number of detected events in the experiment. To circumvent this problem we use an event merging procedure: during digitization we produce so-called summable digits. These are digits before the addition of electronic noise and pedestal subtraction. Adding the summable digits from background and signal events produces merged events. Each background event is used n times for merging. Since the simulation of the background events dominates the computing time, in this way the event statistics are increased by a factor of n.

A similar technique called embedding consists in mixing data from simulation with real events. This allows for a realistic evaluation of track reconstruction performance in a high-particle-density environment. Here, events are mixed at the level of ADCs and are subsequently passed to the standard reconstruction code.

The current ALICE software framework based on AliRoot 5.0 supports embedding and event mixing use cases and provides a base for the implementation of parameterized fast simulation. It is clear that we will have to make use of these alternative simulation techniques more than we did so far in order to compensate for increased CPU needs when using the GEANT 4 transport engine.

6.1.3 Reconstruction

The reconstruction proceeds in two steps. The first consists of converting the raw data to space-points (clusters) for each detector separately while in the second step the track finding and fits are performed.

The tracking in the central barrel starts with the determination of the interaction vertex using the two innermost layers (SPD) of the ITS. After preliminary vertexing, the track finding and fitting is performed in three distinct stages: inward-outward-inward. The first stage starts with finding the TPC tracks in two passes: using the seeds with two clusters and the vertex constraint, then with three clusters without constraint. The seeds are used as input for the Kalman filter propagating inward. After the reconstruction in the ITS all tracks are prolonged to their point of closest approach to the preliminary interaction vertex, and the outward propagation starts. The tracks are refitted by the Kalman filter in the outward direction using the clusters found at the previous stage.

Once the track reaches the TRD (R=290~cm), an attempt is made to match it with one with TRD tracklets (vectors made of clusters presumably produced by the same particle) in the six TRD layers.

Similarly, the tracks reaching the TOF detector are matched to the TOF clusters (the track length integration and time-of-flight calculation is stopped at this stage) and then the tracks are propagated further for matching with space points in the EMCal, PHOS and HMPID.

At the final stage of the track reconstruction all tracks are propagated inward starting from the outer radius of the TRD. In each detector (TRD, TPC, and ITS) the tracks are refitted with the previously found clusters.

Work is currently in progress on using the TPC tracks reconstructed by the HLT as seeds for the offline reconstruction, which will allow us both to speed it up and to reduce the memory consumption. Coupled with the planned improvement in HLT reconstruction this will allow us to reduce the offline TPC reconstruction to an afterburner for the low p_T tracks (if not to completely eliminate it).

Another target we want to achieve is to use the TRD not only for the PID but also for a more precise measurement of high p_T tracks. Currently the residual misalignment and mis-calibration prevent us from using the TRD for the update of the track kinematics. This is supposed to be solved by means of the calibration and alignment procedure involving all tracking detectors simultaneously. The corresponding framework based on the MillePede approach (already successfully used by Alice for individual detectors) is yet to be implemented.

Finally, we will attempt to speed up the ITS reconstruction both in the TPC-ITS prolongation and in standalone mode by the algorithms being developed for the upgraded ITS detector (planned to be used after LS2).

Based on these achievements and the demonstrated tracking performance of the HLT system, several possible further improvements by utilizing the online reconstruction results are currently studied. An algorithm to further reduce the data size on tape by removal of identified background clusters, i.e. clusters not from collision tracks is being evaluated.

6.1.4 Data analysis

We distinguish two main types of analysis: scheduled analysis and user level analysis. They differ in their data access pattern, in the storage and registration of the results, and in the frequency of changes in the analysis code.

6.1.4.1 User level analysis

The user level analysis is focused on a single physics task and typically is based on filtered data from scheduled analysis. Each physicist may also access directly large parts of the ESD in order to search for rare events or processes. Usually the user develops the code using a small subsample of data, and changes the algorithms and criteria frequently. The analysis macros and software are tested many times on relatively small data volumes, both experimental and Monte Carlo data. The output is often only a set of histograms. Such tuning of the analysis code can be done on a local data set or on distributed data using Grid tools. The final version of the analysis will eventually be submitted to the Grid and will access large portions or even the totality of the ESDs or AODs. The results may be registered in the Grid file catalogue and used at later stages of the analysis. This

activity may or may not be coordinated inside the PWGs, via the definition of priorities.

While this kind of analysis was always seen as the least efficient one, we do not wish to ban it in the future as it allows individual physicists to carry out early exploratory work. Instead, we plan to better monitor such activities and trigger warnings sent to job owners and PWG conveners as soon as inefficient use of resources is detected. We also plan to continue investing in the development of scheduled analysis tools aiming to make it even more attractive and easy to use so that most users naturally migrate to that framework.

6.1.4.2 Scheduled analysis

The scheduled analysis typically uses all the available data from a given period, and stores and registers the results using Grid middleware. The AOD files, generated during the scheduled analysis, can be used by several subsequent analyses, or by a class of related physics tasks. The procedure of scheduled analysis is centralized and can be considered as data filtering. The requirements come from the Physics Working Groups (PWG) and are prioritized by the Physics Board taking into account the available computing and storage resources. The analysis code is tested in advance and released before the beginning of the data processing.

In addition, ALICE runs organized analysis using so-called analysis trains. Analysis trains group analyses of different users together that process the same dataset with the following advantages:

- The total overhead resource usage that is associated with the execution of each analysis code separately is significantly reduced. This includes for example job management, input & output file management, job-start overhead and the resources consumed for the decoding of the input data;
- Users have to execute much less Grid operations by themselves and save time;
- Due to a number of automatic systems dealing with failures and merging, the total turn around time is less than achievable by a single user;
- Due to tests before the submission of an analysis train, wrong configurations or faulty code are spotted, reducing the number of jobs that would fail compared to single users submitting their jobs.

Analysis trains are organized on the level of the Physics Working Groups (PWG) in ALICE und steered by up to three operators per PWG.

Composition, testing, and submission of trains are steered with a dedicated web front-end. On the one hand, this front end allows the users to configure so-called train wagons that specify the analysis code that they would like to run on certain data sets. On the other hand, operators use the front end to test train compositions that evaluates the resource consumption per train wagon (CPU time and memory) as well as issues warnings in case of memory leaks or excessive output. Only after a successful test the train can be launched.

Typically each PWG has about 6 different trains (different collision systems or data type like data or MC). Trains have between 5 and 100 different wagons and

run on datasets of up to 50 TB. Each train is launched twice per week in sync with the deployment of the analysis code. In this period the number of concurrent running jobs from analysis trains peaks up to 20 000 which is about 40% of the total available ALICE resources. However, the averaged number of concurrently running jobs is about 5000, corresponding to a resource consumption of about 400 CPU years per month. These Grid jobs are given the highest priority among all jobs in ALICE. The turn-round time of a train (between submission of a train and the point in time where the output has been merged) is between 24 and 72 hours depending on the configuration.

The train system was created at the end of 2011 and since then has replaced a significant amount of separate user analyses. ALICE is satisfied with the success and growth of the system. At the same time room for improvement can be seen which should be addressed in the future. In particular,

- We aim to reduce the train turn-round time to less than 12 hours allowing a work cycle where trains are launched in the evening and results are available the next morning. In particular room is seen to optimize the merging step where a small fraction of failures in the merging process can cause significant delays. We also work on active data management procedures that aim to consolidate datasets that are used by active analysis trains onto fewer sites where sufficient processing capacity exists in order to optimize analysis train performance;
- We aim to increase the code distribution frequency, potentially daily. In this scenario, it might be necessary to revise the policy that trains are submitted per PWG to prevent too many small trains to be started on a daily basis;
- We would like to extend the system to allow the creation and processing of reduced data sets ("tuples") including the required bookkeeping.

6.2 ATLAS

The target for data reconstruction is to achieve a speedup factor of two to three with respect to the current software performance, i.e. to achieve an average reconstruction time per event in high energy running with 25-ns bunch spacing at μ=25-40 close to that in the 2012 data taking. The envisaged improvements are to be achieved by targeted algorithmic improvements as well as introducing (auto)-vectorisation into as many areas of ATLAS code as possible, thereby gaining factors of speedup by utilizing the current CPU architectures in an optimal way, and introducing modern, more efficient, linear algebra libraries into the ATLAS code base.

As one of the first of these libraries, Eigen is implemented in the offline reconstruction software; most changes will be in the tracking domain. Improvements are expected due to its vectorized implementations of matrix and vector operations, but more due to its more modern implementation that allows the optimizer to further improve the generated code. Modern compilers are a pre-requisite, and over the last year, ATLAS moved subsequently to modern versions of gcc and will also investigate other compilers. Further improvements will be mostly done by vectorization, improving memory layouts to allow faster access and algorithmic improvements. Smaller improvements can be expected by some other library replacements, e.g. the math library (can be as much as 10%) and memory allocators (with a possible penalty on total memory consumption).

In order to reduce the memory requirement, ATLAS is introducing the multi-process event-level parallel version of the ATLAS software framework (AthenaMP), which takes advantage of shared memory by utilizing the copy-on-write feature of the Linux kernel. As the next step, ATLAS software will be moved to a new concurrent framework with both event-level and algorithm-level parallelism using modern threading techniques and libraries such as the Intel TBB. The effort is planned to be shared together with other LHC experiments and should also result in ATLAS software with a low enough memory footprint to be ready for the future many-core architectures, as well as potential use at High Performance Computing facilities and IaaS resources. The work on the new framework started at the end of 2012 with the aim of fully introducing this framework during LS2. However, if memory usage during data-taking in Run-2 rises too high and causes severe CPU penalties by e.g. being able to schedule jobs only on a subset of the available CPUs to gain memory, ATLAS plans to introduce the concurrent framework in the offline reconstruction already during the data-taking of Run-2.

Similar plans are also in place for the full (Geant4) and fast (Geant4 combined with parameterized response) simulation, pileup simulation and digitization and simulation reconstruction.

Between Run-1 and Run-2 the ATLAS software was also switched from 32 bit for Run-1 to 64 bit for Run-2. This gave in general a speed-up of some 20% at a cost of 50% higher memory consumption. After the switch, it has become important to again change the focus on memory optimizations.

This also means that before athenaMP comes fully into production, we can

expect higher memory consumption at grid sites for production jobs. Despite higher pileup, ATLAS aims at keeping the RSS to about 3GB in Run-2, while it currently it goes well beyond that threshold with the Run-1 software setup.

As a parallel target, the AOD event sizes should be optimized to the values of 0.25 MB/event (the 2012 value) for μ=25 (and 0.35 MB/event for μ=40) pileup conditions in 2015, which compared to the current software performance requires again a considerable event size optimization. The objective is to achieve this by technological improvements, reducing the physics content only as the last resort to remain within an achievable computing resource budget. Analogous improvements are planned for the other data formats, i.e. RAW, ESD, simulated HITS, *etc*. The very first step will be replacing the compression algorithm, which can yield a small size reduction with no loss of information. Further reduction should aim at removing redundant information, which is still present in AODs.

The campaign to increase simulation speed further during LS1 is already underway. Considerable effort is being invested in developing a new simulation framework (ISF), which will enable mixing fast (parameterized) and full simulation options, selectable at the particle level, and in speeding up the pileup simulation (digitization). Another promising avenue being investigated is the possibility to use zero-bias events from real data to accurately describe the actual pileup when overlaid with simulated physics processes. The viability and speed gains of this approach in pileup digitization are at present still being investigated.

The full Geant4 simulation of the ATLAS detector is very CPU intensive due to the design of the ATLAS detector, in particular the Geant4 simulation of the LAr sub-detectors takes on the order of 60% of the total CPU time per event. ATLAS software experts are intensively working on speeding up the Geant4 processing time further and managed to speed up the processing time in the Run-1 period by about 30% already. Work to further reduce the CPU time for Run-2 is continuing and further gains are envisaged, e.g. by introducing faster random number algorithms and close collaboration with the Geant4 experts.

To alleviate the CPU demand of full simulation, ATLAS has in Run-1 introduced a fast simulation called ATLFAST-II, which parameterizes the calorimeter response and thereby gains an order of magnitude in speed. This has been shown to carry a cost of reduced precision, which has been carefully evaluated. It is being used for a large set of analyses, in particular for New Physics searches. At the end of Run-1 the ratio of fast simulation to full Geant4 simulation reached the ratio of almost one to one. ATLAS has a dedicated team working on improving the fast calorimeter simulation for Run2. The Run1 version of this simulation is limited by the hadronic shower parameterization modelling. Due to these limitations, the modelling of the identification of hadronic decays of taus (and the background rejection against QCD jets) is not accurate enough for many physics analyses. In addition, jet sub-structure observables, which rely on accurate modelling of calorimeter clusters, are not simulated accurately enough for many physics analyses. The improved parameterizations expected for Run2 should address these issues, and allow essentially all final states to be accurately simulated with fast simulation, although for ultimate precision in some analyses, full simulation will still remain the preferred choice.

In addition, for Run2 ATLAS simulation experts are developing further flavours of even faster simulation, e.g. combining the ATLFAST-II with the parameterized tracking (FATRAS), which speeds up the simulation time by a further order of magnitude. Also investigated are the possibilities of simulating different 'regions of interest' or specific particles with different levels of precision, which can introduce further substantial reductions in CPU consumption. All the simulation types are now integrated into a new Integrated Software Framework (ISF) and are being commissioned for Run-2.

An ultra-fast simulation chain, involving special fast simulation and reconstruction by developing 'truth-assisted' algorithms is also being envisaged. These algorithms could very much reduce the total ultra-fast simulation time, and would, due to the small CPU penalty for re-runs, enable one to avoid the storage of all intermediate steps (HITS, AOD) and thus to store samples directly in final (skimmed/slimmed/thinned) group analysis format only. This possibility is as present in the prototyping stage and is expected to be commissioned during early 2015.

6.3 CMS

CMS has started a program of work aimed at optimizing both the reconstruction and simulation applications. These optimizations enter into production releases in a 6month cycle of developments which allow us to both snap-shot the current technical performance gains as well as test for any regression in physics performance at a larger scale and with a more thorough involvement of collaboration validators. The current schedule of releases out to the start of data taking in 2014 is shown in Table 31.

Table 31: CMS Release Schedule

Release	Scheduled Date	Primary goals
6_2_0	July 2013	Release for first 13 TeV / Run 2 detector samples
7_0_0	November 2013	Multi-threaded CMSSW framework Geant 4.10 and Root6 builds done for parallel validation
7_1_0	July 2014	Release for MC physics samples for startup analysis Geant 4.10 and Root6 are default
7_2_0	November 2014	Baseline release for HLT in 2014 run
7_3_0	April 2015	Bug fix release To be used for offline reconstruction of first data

Note that the above table implies a plan for integrating and testing major new versions of the common projects, Geant4, and ROOT6. Variations of the 7_0_0 CMSSW release will be made and compared with results from the same release with the older versions of these tools. We then plan to work closely with the developers of these products to resolve any issues we find over the next 6 months of 2014 in order to be able to move to the new versions by July of 2014.

Over the course of run 1, CMS was able to improve the performance of the reconstruction by a factor of 10 for the pileup seen in run 1. However simulation of high luminosity, and therefor pileup conditions, expected in run2, showed that the high pileup behaviour of these algorithms were highly non-linear. The CPU time per event of the legacy run 1 release, 5_3_X, at 1.5E+34 with 50ns bunch spacing was 175sec/event. Technical improvements in the code since then have reduced this to 107 sec/event as measured in the 6_2_X release and shown in figure 2 below. The CPU time per event at low pileup, 0.5E+34 has not changed since 5_3_X, so it is really the control of the combinatorics that has improved. In the figure below, there are two curves for each bunch spacing scenario. The curve labelled postLS1 is a tuned configuration that is still under study but shows promising technical performance improvements. This configuration has

been shown to practically not change the physics performance of the code for the bulk of CMS analysis, but it is operationally more challenging for the detector group. It requires a stability and correctness of the tracker cluster charge calibration that has not been required before.

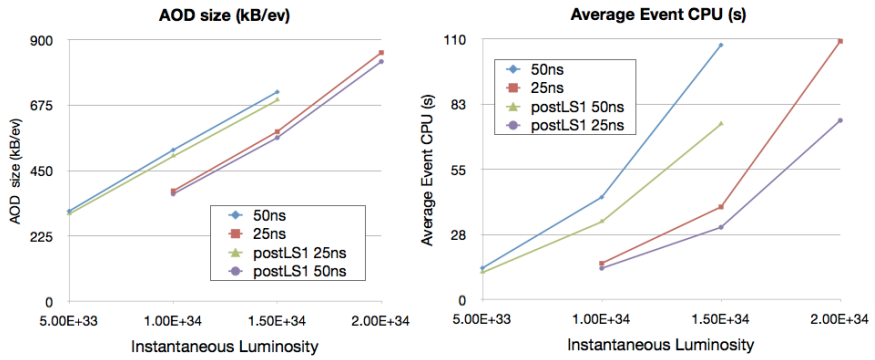


Figure 41: Performance metrics for the latest CMSSW release

Further improvements in the reconstruction are still being developed in the 7_0_X series.

Work on the performance of the simulation is also on going. CMS has invested effort in implementing the Russian Roulette[5] approximation technique into its FullSimulation and plans to deploy it and other mathematical function optimizations in the 7_0_X release. Detailed studies on the simulation of high pile-up conditions seen in 2012 have shown that a large window of time slices needs to be included in the digitization in order to reproduce the response of the electromagnetic calorimeter, whose reconstruction has proven to be extremely sensitive to this parameter. This implies 400 minimum bias events will have to be mixed to create <25> pileup samples and 2240 minimum bias events for <140> pileup samples, needed for HL-LHC upgrade studies. This will create a heavy I/O load for computing and is something that should be done infrequently. We have therefor redesigned the workflow and implemented what we call a "pre-mixing" scheme. First a large sample of zero bias, e.g. pure pile-up events, with the correct pileup mixture is created by superimposing single minimum bias events at a facility with large I/O capacity. Then each single pre-mixed pile-up event is superimposed on a signal event, in a one to one ratio that is much less challenging for I/O. The same pile-up sample can be reused on top of any signal dataset, which avoids wasting CPU to reproduce it for each new sample since large pileup digitization is becoming a significant fraction of the total simulation CPU time.

During long shutdown one, CMS is redesigning its framework to support coarse and fine-grained parallelism in order to operate efficiently on multi-core architectures. It will be able to process multiple events including events across

[5] https://twiki.cern.ch/twiki/bin/view/Geant4/Geant4PerformanceTips

run or luminosity block boundaries concurrently as illustrated in Figure 42 below.

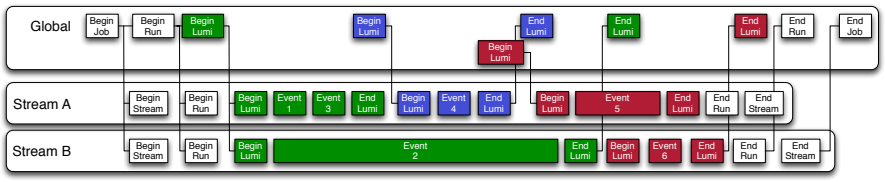


Figure 42: Multi-threaded CMSSW design

Within a Run, Lumi (a Lumi is a subset of the run in which beam and calibration conditions are constant) or Event, it will run multiple paths and modules concurrently based on what they have declared as their input dependencies. Modules that produce event products will be called on demand by the filters and analyzers that request their data For each event there is a synchronization point between normal paths, which can contain filters, and end paths, which are primarily used for output modules. This is illustrated in the Figure 43 below. This project has been broken up into 3 phases the first of which will be delivered in the November, 7_0_0 release. In that phase event level and fine-grained parallelism within a module will be supported. Later phases will be implemented in 7_1_0 supporting modules processing within an Event in parallel and 7_2_0 supporting processing Runs and Lumis in parallel.

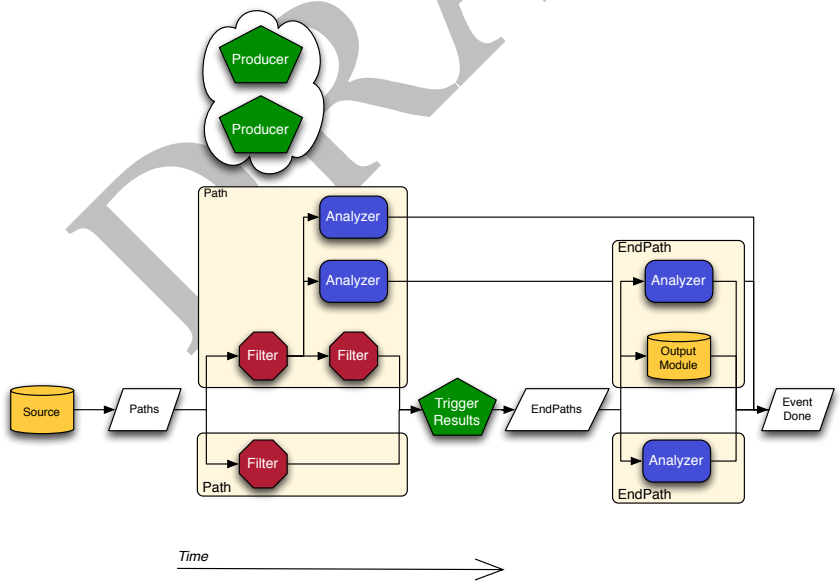


Figure 43: Illustration of sub-event parallelism

6.4 LHCb

6.4.1 Systematic performance measurements.

In order to monitor and improve the performance of all of its software, LHCb started some time ago a program of systematic regression and performance testing of its main applications. The first step was the implementation of a framework allowing to gather and compare quality and performance metrics: this was done during 2012/2013 and the LHCb Performance and Regression testing framework, LHCbPR[6], is now available for use by applications.

While the LHCbPR framework is still under development, the second phase to identify use cases representative of the various productions phases, was nonetheless already started. This phase is obviously critical, as a good coverage is critical to ensure a valid monitoring of the quality and performance in place. These efforts already started yielding results, showing the reliance on mathematical libraries from GLIBC/libm, but they also put in evidence the quality of the optimization efforts performed until now: obvious bottlenecks have already been resolved. Performance issues are being addressed both by a bottom-up approach, optimizing algorithms starting with the most time consuming, as well as by a top down approach, improving the Gaudi framework wherever possible.

Improvements have to be considered application by application and can be made either at the algorithmic level, at the data level (e.g. Fast Simulations in Gauss using simplified models), or at the technical level: a number of studies are under way to evaluate the impact of new compilers (GCC 4.8, Clang, ICC) as well as compilation options (optimization levels, auto-vectorization. etc.).

The LHCb Software currently under-uses vector registers, and significant gains could be made if this was improved: while explicit vectorization is possible, it leads to complex and difficult to maintain code. Auto-vectorization by compilers is an appealing solution but studies on LHCb code have shown that it does not yield results unless the code and data layout in memory are re-arranged. Studies are therefore under-way to evaluate this approach.

For example, the Brunel code performs rather well when measured with profiling tools such as the Intel perfmon tools. For example Brunel achieves 0.97 instructions per cycle and 2.3% cache misses; however there is still room for improvement if the code can be rearranged to make better use of vector instructions.

Of course longer term studies are also under-way, e.g. the evaluation of other CPU architectures (ARM for Online), or of the use of GPUs, and in all cases the LHCbPR framework should allow to validate the quality and the performance of the improved software.

A major concern for LHCb is that, to have a major impact, such optimisations require dedicated effort with advanced computing skills, which is not available in the collaboration. We suggest that the funding agencies should fund a limited

[6] https://twiki.cern.ch/twiki/bin/view/LHCb/LHCbPR

number of computer scientists to make the necessary skills available, as this would in the long term lead to savings in the computing resources required for data processing.

6.4.2 Reconstruction

The LHCb application that places greatest constraints on CPU requirements is the reconstruction program, Brunel, since this is used to process all real data events at least once. CPU consumption by Brunel dominates the time required for a reprocessing campaign, and defines the size of facilities required for the prompt processing rate to keep up with data taking. For this reason, studies of software efficiency have concentrated on the Brunel application.

An important feature of the LHCb reconstruction is that large parts of it execute the same code that runs in the HLT farm. Thus code optimisations that were done to improve the latency of the HLT have directly benefited the Brunel code, resulting in a code that, from an algorithmic point of view, was already highly optimised from the onset.

On the other hand, the reconstruction software was originally designed and tuned for an average number of visible pp interactions per bunch crossing (mu) of 0.4[7] whereas the real LHC machine conditions have led LHCb to operate at mu ~1.8 to achieve its physics goals. Software optimisation efforts have therefore concentrated on ensuring that the software is robust against pileup or, equivalently, mean event track multiplicity. The figure below shows that the measured reconstruction time per event as a function of track multiplicity increased linearly with track multiplicity, up to multiplicities in excess of 200 tracks per event, corresponding to mu ~ 3.5. Assuming operation with 25ns bunch spacing, the expected mu in 2015 is **lower** than in 2012 up to the largest instantaneous luminosities that can be tolerated by LHCb due to radiation considerations, giving us confidence in the robustness of our estimates for the CPU requirement for reconstruction in 2015-2018.

Possible future optimisations include the use of faster approximations to time consuming mathematical functions, and improvements in the vectorisability of the code.

We are particularly interested in optimising the code against accuracy of math libraries. As an example, a recent change in the libm library distributed by default on slc5, intended to fix some rounding error problems, led to a slow down of up to a factor 5 in various mathematical functions, including exp, pow, sin and tan. On the other hand, a recent study[8] has shown that Brunel results are robust against accuracy changes in the math library, in particular when using the approximations to mathematical functions coded in the VDT library; a global replacement of libm by VDT led to a 3.5% speedup of the Brunel event loop.

[7] LHCb Computing TDR LHC-2005-019; Reoptimized Detector TDR LHC-2003-030

[8] https://indico.cern.ch/getFile.py/access?contribId=17&sessionId=3&resId=0&materialId=slides&confId=236650

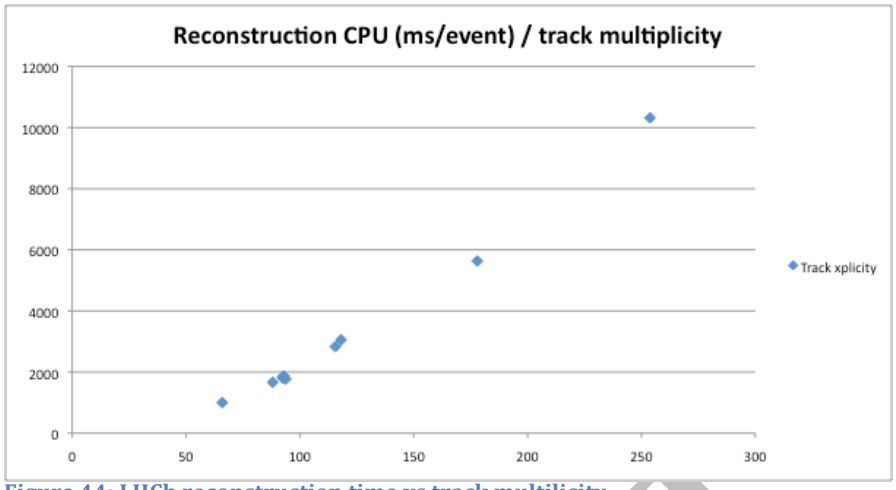


Figure 44: LHCb reconstruction time vs track multilicity

6.4.3 Simulation

The LHCb simulation currently places great constraints on disk space as all events processed through the detector simulation, reconstruction and stripping are normally saved in an MC-DST format, where the RAW and DST event are saved together with the associated MC truth. Considerable effort has been spent in the last few years in reducing the size of the MC samples while providing all necessary information for the physics studies. This has been achieved by processing only events satisfying tight cuts at the generator level or saving only events selected by specific trigger and stripping lines, depending on what most effective in term of MC sample size for given analysis. A very recent effort has been in developing an MC-microDST similar to the MDST of real data that is under validation for use by the physics community.

The CPU time used by the LHCb simulation application based on Geant4, Gauss, is expected to increase in the future due to higher precision of the modelling of the physics processes used as they are being tuned to better match the very high precision measurement of the experiment. The CPU time of the simulation is also expected to grow after LS2 due to the increase in experiment Luminosity with mu expected to raise to ~ 5-6. For this reason an optimization campaign of Gauss is starting in combination with its tuning to find the optimal solution of very high precision modelling and CPU time of the overall application, based on the choice of Geant4 cuts and LHCb implemented transport cuts. Different levels of fast simulations are under consideration with as first step a full benchmarking of the simulation to identify the hot spots. The LHCbPR framework will provide the infrastructure to carry out this work. With the LHC operating at 25 ns bunch spacing the simulation will need to address also the out of time pileup. A fully detailed model exists but while fully appropriate for detailed studies is not CPU sustainable for the sample sizes needed for physics studies. Alternative solutions will be considered for in depth investigation.

7 Distributed Computing

It is anticipated that LHC computing will continue to use a worldwide distributed computing model for the foreseeable future, driven by considerations of available funding, how that funding may be directed, and the non-negligible advantages of having computing centres located close to the physics groups. This last has been an important factor in building trust within the LHC user community that computing resources in this global system would actually be available to them, when they needed them. This factor is still important in being able to attract new resources, particularly from less well-connected regions of the world.

However, technology has evolved in the last 10 years since the LHC computing models were first designed. Today it is commonplace for people to access compute services over the network (for example the wealth of cloud services everyone has access to from Amazon, Google, and many others), and the need for "ownership" of resources is probably less strong now, particularly as (academic and research) networks in many countries now provide significant bandwidth to universities and national labs. In addition, it is far from clear today that building many small Tier 2 (or Tier 3) sites, is the best and most cost effective use of the limited available funding. Thus we should anticipate some evolution in the ways that countries may want to provide their computing and storage resource pledges. As the costs of cloud computing decrease over the coming years, and as the scale of computing for LHC continues to grow (even within limited budgets), there may be opportunities for economies of scale to provide additional benefits:

- Several small Tier 2 sites could decide to locate all resources in a single physical site, and gain through reduced operational effort, potential benefits in procurement via larger single orders, acquiring significant network bandwidth at a single site rather than at several, etc.;
- Funding agencies may want to consolidate resource provision across sciences, or between different types of computing need (HPC, high-throughput, etc);
- Eventually the cost of commercially procuring computing for LHC may become competitive with the in-house costs, particularly as full cost accounting is implemented in university departments. Thus we may anticipate eventually that some pledge capacity could be provided through commercial contracts, presumably in the form of cloud services.

Given this kind of evolution in the environment, it is probably more useful now to specify the needs for computing in terms of the tasks that need to be done, and to be less prescriptive in terms of the functions of the traditional Tier 0, Tier 1, Tier 2 model that we have had so far. Indeed, the experience with LHC data over the past 4 years has already led to some pragmatic changes in the experiment computing models, where the original functions of the various Tiers have become blurred, in order to simply obtain the optimum value from the available resources, rather than blindly adhering to the functions specified in the computing models.

This kind of consideration is also important as (hopefully!) new sites and new resources continue to be attracted to the LHC computing effort. For example, at the moment a new large site that is of the scale and intending to deliver a high quality of service would be considered as a Tier 1, which implies that it must also provide tape archive facilities. However, in some cases, it does not necessarily make sense for the experiment to have additional archive sites in this way, and may be more important to have a large, very well networked, data distribution centre or compute facility. For the coming years it is important to introduce this additional level of flexibility.

Finally the possibility to make use of opportunistic resources efficiently, which could be an important source of certain types of processing power, is also becoming more important. This has implications for the requirements that the experiments place on the sites that they use, and the complexity of software and services that need to be deployed at sites in order to make use of them. Clarifying this from a functional point of view will help in understanding how to best make use of such resources, in particular if compute resources are offered, but not storage capacity (other than temporary cache space).

7.1 The main use cases

In order to clarify the set of computing functions that the sites need to provide, it is instructive to review the basic processing activities that the experiments need to execute.

1. Calibration and alignment. This is a set of functions performed at the Tier 0 today, which requires rapid turn-around and feedback to the detector, and the conditions database. It is a highly structured and organised batch processing activity, under the control of the core computing teams on behalf of the experiment. The activity requires the availability of very recent data samples, and adequate connectivity to the experiments' data acquisition clusters. It is fairly labour intensive and requires the oversight of the core teams.
2. The reconstruction and reprocessing activities today happen at the Tier 0 and Tier 1 sites, and requires access to the raw data, and the recall of that data from the tape archives. Practically these activities need to be rather close to the archives in order to avoid large network data transfers. Again these are organised batch activities, managed by the computing teams.
3. Stripping, creation of AODs, derived data sets, etc. Depending on the experiment this may require a complete pass through the full processed data sets, or be created as output from the processing itself. In any case it requires access to the full processed data set, and is also typically done where there is sufficient storage, in some cases also needing access to the tape copies. Again these activities typically today are done at the Tier 1 (and Tier 0) sites.
4. Simulation. Both the activities of event generation and the full (or fast) simulation steps are rather compute-intensive and have relatively little I/O needs. These activities need compute clusters, with enough storage to cache the output while it is staged elsewhere for longer-term storage or further processing. This workload has made use of Tier 2 sites, and has also been successfully deployed on opportunistic resources such as cloud

or HPC systems with little or no local storage, the input and output data being staged directly over the network to a site capable of the data serving. Subsequent steps of the simulation workflows (reconstruction and analysis) follow the same requirements as for real data.

5. Organised or group analysis. Typically runs at Tier 2 sites, requiring the input physics data sets to be present. Those sites clearly need sufficient storage capacity to enable this; and that capacity must scale with the size of the compute resource. It is also important that the WAN connectivity of the site also scales with the size of the compute and storage resource. Data has to be placed at the site, preferably close to when it is required, and this needs adequate network bandwidth. Also the results of the analysis work must be served to other sites, again needing adequate bandwidth.
6. Individual or "chaotic" analysis. The tendency is for this type of work to happen on local compute resources, which may range from the scale of individual laptops or desktops, to a small cluster, to a large national analysis facility. While that resource is mostly outside the scope of the WLCG, the requirements on being able to serve data to those resources falls squarely within the scope. Thus Tier 2s or "data serving" sites must have large reliable disk storage, and significant network bandwidth in order to manage the data traffic.

7.2 Functional Tasks

With the main use cases as discussed above in mind, we can set out the main functionalities that the sites contributing to WLCG should provide. In future we may expect that sites would define their roles, and consequently the type and scale of resources that they provide, based on these functionalities, more than simply a label of a Tier.

Some of the key functions that are necessary include:

1. Data archiving (on tape for the foreseeable future). This function is a long-term commitment of 20 years or more, and requires a site with experience in mass storage and tape systems. Data must be actively curated, migrating between tape generations, drive, and robot technologies, as well as providing a regular control that the data is still accessible. Data losses, which at some level are inevitable, if only due to technical failure, must be recovered from copies at other sites. A data archive requires a disk buffer at least large enough for the expected data migrations to and from the archive that it must manage, and adequate network bandwidth, at least 10 Gb/s redundant paths.
2. Large-scale data serving. A site capable of holding significant data sets online on disk and serving them over the network to other storage or compute resources. This would also require significant expertise in the management of large storage systems, at a very high level of reliability and availability. Such a site would require significant network bandwidth – again multiple 10 Gb/s connections as a minimum.
3. Batch compute facilities. Here there may be two different requirements, based on the amount of I/O required.

   a. For reconstruction/reprocessing, batch analysis type of work, the compute cluster must be matched by large local storage resources, which can easily be populated from an archive facility or other data-serving site. Consequently it will also need adequate network bandwidth. Both the storage and network bandwidth must be scaled adequately together with the size of the compute cluster.
   b. For simulation type (relatively little I/O) of work, little or no permanent local storage may be available (as in some opportunistic resources). As long as the network connectivity is sufficient to manage the scale of remote I/O to a storage site such a site can be useful. Alternatively, a site that may be poorly connected, but with a reasonable amount of local storage, could also be useful for this type of non-time-critical work.
4. Infrastructure or experiment central services. Some sites are required to run key services for the WLCG or specific experiments. These include the key database services, FTS services, workload managers (such as Panda, Dirac, etc), VOMS, and a few others. These all require a very high guaranteed level of availability and reliability, since they are usually critical for the operation of the infrastructure or the experiments. Sites providing these must be capable of providing a high quality of service, and must be very well connected to all other sites. Today these services run at the Tier 0 and Tier 1 sites.

7.2.1 Functions of the Tier sites

Given the above discussion, we may expect to see some evolution of the functional roles of the various tiers. The following indicates the expected evolution:

- Tier 0: functionality essentially as now, although in shutdown periods will provide more Tier 1-like and Tier 2-like functions;
- Tier 1: most of the prompt reconstruction, reprocessing, and analysis activities; Data serving;
- Tier 2: simulation production, user analysis, as well as MC reconstruction tasks;
- Archive (tape) sites: usually but not necessarily co-located with Tier 1 sites (the implication here is not that existing Tier 1s will stop providing this service, but that new sites, otherwise providing the scale and quality of Tier 1 services, may not need to provide additional archive space);
- Tier 0+Tier 1 sites are expected to run central services (databases, CVMFS stratum 1, FTS, etc); and to provide 24/7 levels of support; while Tier 2 sites would not run such services and would provide only business hours support levels.
- Opportunistic resources would be essentially only useful for compute intensive tasks depending on their actual connectivity and the possibilities of data transfer in and out.

From these arguments, it is clear that the use of opportunistic resources, and the attractiveness for smaller Tier 2 sites to offer resources, will be greatly aided by aiming to:

- Reduce the need for site configuration to as little as possible, through mechanisms such as the use of standard ways of accessing resources, download of software through web caches such as CVMFS, etc;
- Avoiding the need for any long-term state at the site: tasks should clean up as they finish, and data produced should be transferred out to long term storage.

Such strategies will reduce the overhead of site management for WLCG needs to very little, for the class of sites where available effort is limited. Thus we may agree that there are slightly different styles of Tier 2 sites – those that have long term state (i.e. a managed storage service), and those without long term state (data caches only). The latter would be a model for opportunistic resources also.

7.3 Networking

From the above discussion it is clear that all sites are expected to have excellent network connectivity – the scale of which rather depends on the specific scaling of the services. For sites where the connectivity is unavoidably limited, there are nevertheless functions that can still be usefully run there.

As has been observed, the original design of the grid for LHC tended to make pessimistic assumptions about the available bandwidth and likely reliability of the wide area networks. However, the reality has been quite different, with increases in bandwidth often exceeding expectation, and with an effective reliability (due to both technical quality and planning for redundancy) that is extremely high. This has already led to significant evolutions of parts of the experiment computing models, and to a simplification of the deployment modes of several key services.

The performance of the network has allowed a more flexible model in terms of data access:

- Removal of the strict hierarchy of data moving down the tiers, and allowing a more peer-peer data access policy (a site can obtain data from more or less any other site);
- The introduction of the ability to have remote access to data, either in obtaining missing files needed by a job from over the WAN, or in some cases actually streaming data remotely to a job.

These changes, together with more intelligent policies of placement and caching of data, have improved the use of the network capacity, and most importantly, increased the efficiency of data processing tasks.

In the original computing models, there were a number of services that were replicated at many sites, and some of these were particularly complex (e.g. the file catalogue databases, and some of the conditions databases). A combination of running only a central service, and simple caching technologies using e.g. Frontier and squid caches, has simplified the deployment, and thus support models. Similarly the complex deployment of the LFC file catalogues (particularly for ATLAS) has been consolidated to a single central instance at CERN with a replica at BNL.

As was noted in an earlier chapter, network bandwidths are expected to continue to increase over the coming years, with the prices for sufficient capacity

for anticipated LHC needs dropping. Providing adequate connectivity to most sites should be easily feasible, and in many cases in Europe and the USA upgrades of connections to Tier 1 and Tier 2 sites is already happening, and many such sites will soon have 10 Gb connections or better. While it is clear the WLCG must plan for the network capacity that it requires, it is likely that those needs will not require more than the capacity; for example there will be no need to make use of software defined networks, or other complex technologies, in order to support LHC computing.

8 Computing Services

In 2011 WLCG set up a series of working groups to look at the technical evolution of the various aspects of the grid implementations. These "Technical Evolution Groups" (TEG) produced a set of reports[ii] in early 2012 setting out the likely directions and needs covering these aspects. The groups covered the following topics:

- Storage and Data Management
- Workload Management
- Database Services
- Operations and Tools
- Security

Already in 2010 the WLCG had looked at the evolution of data management recognising that the early experience with LHC data showed that changes in strategy would be needed. Input from that workshop fed into the TEG reports. In this chapter we extract the main conclusions from that previous work, and outline some potential strategies. In many areas of implementation of these different aspects of the WLCG grid, the LHC experiments are now converging on common solutions (or common types of solution). This convergence will hopefully simplify the set of software that must be deployed, operated, and supported, ultimately hopefully reducing the overall operation and support cost.

Another theme that is coming to the fore is the desire to use solutions that are not specific or peculiar to HEP, where this is realistic. Use of standard software, protocols, and tools where possible again helps to make the WLCG infrastructure and services easier to maintain, operate and evolve, as long as such standard solutions can satisfy our requirements. On the other hand there are clearly areas where LHC is unique – such as the need for a globally federated data distribution and access service. However, even in such an area LHC may be unique today, but there are many other sciences that will soon have similar problems to solve, and so HEP should ensure that tools and services that we develop might reasonably be eventually made available to those other communities.

In the following sections we outline the main areas of technical implementation of the grid infrastructure, and discuss the possible evolutionary paths, and mention where work is on-going or needed.

8.1 Workload Management

One of the most significant evolutions over the last several years is the move away from the original grid model of "resource brokering", where a user would contact a service (in the gLite model – the WMS) which determined the "best" place to run a job based on the requirements specified by the user, matching that with descriptions of resources published by the resource providers. This model was essentially an extension to the classic batch service. However, this model has a number of complexities when many sites are involved. For example, there is no simple way of an experiment to communicate to the sites or the brokering service, the relative priorities of different workloads, or to dynamically alter those priorities. In addition, the brokering process relies on a lot of information

to be accurately published by the sites. This was quite complex to do correctly, and in several cases using a round robin brokering was no worse than trying to match sites to requirements.

8.1.1 Move to pilot jobs

Consequently, all four experiments have now moved to the "pilot" model of job submission, where placeholder jobs (pilots) are submitted to available resources, and when such a pilot job starts running it contacts an (experiment managed) central task queue which determines which work load should run next. This model has several advantages, including: the pilot job can in principle better determine its environment and communicate that to help in the task brokering decision, the central task queue allows the dynamic setting of priorities internal to the experiment, that does not need to be communicated to a large number of remote sites. Fears that such central job managers would not scale to the sizes needed are unfounded – with the appropriate scaling of the servers and back-end database infrastructures, today these run at several million tasks per day without problem.

Today there are different pilot job frameworks for each of the experiments using them, while CMS who continued to use the older WMS service, have invested effort over the last year or so to investigate commonalities with ATLAS, particularly as far as analysis tasks are concerned.

It is expected that pilot job frameworks will become the only way of workload submission (apart from some special cases at the Tier 0) during LS1. ATLAS and CMS are working closely together to try and use common components where possible. As ALICE is also planning a significant revision of their computing system, they are also joining that discussion. LHCb, however, has a mature pilot framework (DIRAC) and have no strong reason (or available effort) to change that at the moment.

At a site, jobs are submitted via a standard interface service – the "Compute Element" (CE). This component makes the connection with the local batch service at the site, including a mapping between the grid identifier of the task owner and the local ID to be used at the site, and the connection to the grid level accounting. These CE's are still advertised through the information service, but the experiment workload managers essentially now use that as a service discovery mechanism rather than a dynamic information service.

Since all four experiments now only use (or will only use) pilot jobs for grid work, we could anticipate a simplification of the site entry points (the CE) to a service that is essentially generating and submitting pilot jobs. However, as the existing CE software is stable there is probably no urgency for such a move.

One complexity that has not been satisfactorily resolved is that of fulfilling the need for adequate traceability of workloads. In the standard grid model, the job is submitted with the credentials of the user. This is the case with the pilot, which is run with the identity of the experiment. However, the workload that is fetched from the central task queue can be from any user, and that ownership must be traceable. The existing solution – a tool called from within the pilot (glexec) to switch the identity of the process – has proved difficult to deploy (and ultimately difficult to enforce the use of). The WLCG has insisted that this must

be deployed, in order to address the requirements of traceability, but a better solution is necessary. At the moment the responsibility for trace back if a user workload causes a security problem rests clearly with the experiment; implying that a security issue may result in the entire experiment being banned from a site for some time.

8.1.2 Virtual machines and private clouds

More and more sites are virtualising their infrastructure through various mechanisms. For the moment, the grid services and batch nodes run on top of the virtual machines.

Several sites are also supporting virtual machines through cloud management software. Several experiments are testing direct submission of work to such cloud interfaces. This is useful in order to be able to use opportunistic cloud resources (several such have been offered recently), and for those sites that deploy such software, it may make sense in the future to use those cloud interfaces, rather than through the existing CE's. Thus, we may start to see a gradual evolution from the existing grid mechanisms to private clouds. An interesting aspect of this is that there are large support communities behind such cloud software, while support for the CE software essentially depends on a much smaller community, and has relied on specific additional funding for that support, which may not be there in the longer term.

Virtual machines may also help to improve the efficiency of CPU use, by appropriate provisioning of multi-core job slots, or provisioning the "bare-metal" machine. In either case it becomes the responsibility of the application to optimise the use of the cores available. Various strategies could be available for that, see the discussion in the Software chapters.

8.1.3 Scaling limits

The existing workload managers are fully capable of supporting the workloads requested of them – at the scale of several millions of tasks per day. There is no particular scale issue anticipated for the next few years for these services.

However, for the batch systems the situation is different. At the larger sites, we start to see some scale limits of several of the batch systems. In addition, the support outlook for several of these is not clear, as several of the most popular commercial solutions have been bought recently. Prices, licensing, and levels of support become potentially problematic, particularly for larger sites.

Given the move to pilot jobs, it is not so clear that a full-scale traditional batch system is necessary to support LHC workloads in future. Of course many sites may have other boundary conditions from other supported communities and application needs. However, hybrid models where the cloud management software does coarse provisioning of virtual machines, and simple batch configurations running on top of that where necessary may address some of these scale problems. There are several on-going studies of batch systems and scalability to address these concerns.

8.1.4 Outlook for workload management

Since the demands on a site are now relatively straightforward with the widespread (and almost ubiquitous) use of pilot jobs, it is clear that we may envisage a simplification of the services needed to support submission of tasks to a computing site.

- There will be no requirement for the gLite WMS in future. The remaining use cases will be migrated as soon as possible.
- A CE at a site could be simplified to the key functions: job submission to the local batch system; connection to the grid accounting; logging and "control" of users. If common pilots jobs were used, a site "CE" could be a simple pilot submitter.
- Introduction of cloud interfaces, could eventually replace a grid CE. WLCG should agree a common (subset) of possible interfaces, such as "EC2-like" , or agree to use a common abstraction library, that interfaces to common cloud management software.
    - Experiments need to be able to use such interfaces for any potential opportunistic resources – so since many sites are introducing cloud software, why not just use them directly?
    - Use of VM's will allow whole-node (or multi-core) scheduling in a simple way.
- Reduced required for a traditional batch system (for LHC) with VMs and pilots jobs. Such uses could certainly be simplified, and avoid scaling limitations.

8.1.4.1 Potential future strategy:

- Site gateway could be a simple pilot factory with essential connections to security, accounting, and a switch to permit the site to control the rate of pilots.
- Move away from grid CE's by deploying Openstack or similar cloud management software. Experiments submit directly via cloud interface. This would also allow bursting from one site to another (or to and opportunistic cloud). This also reduces software maintenance load.
- Simplify the use of traditional batch systems.
- Experiments use common pilot jobs, common pilot framework, - improves management at sites by having identical services.
- Common software distribution mechanism – using the CVMFS infrastructures.
- Information system is a means for service discovery and bootstrapping experiment-specific info services, and no longer requires the complex set of information currently published.
- Other strategies (from TEG work), such as CPU pinning, whole-node (bare metal) or multi-core use should be the result of a study on optimising resource use and efficiency, rather than as speculative developments.
- Resolve "glexec" problem: are we happy to accept experiment-level responsibility in case of problems? Should we still insist on sites having user-level control? (Also associated with central banning – VO vs user level).

8.2 Storage and Data Management

Already in 2010 after only a few months of LHC data taking, but with the experience of the previous data challenges, the WLCG organised a workshop to discuss how to manage LHC data in a more efficient and optimal way. This workshop recognised the fact that the network was actually a very valuable resource, that it was possible to reconsider the data models, not to rely as much on pre-placement of data for analysis, and the need to run jobs only where data had been pre-placed. Thus many of the ideas discussed there have already had some positive impact on the way the LHC experiments manage data and data transfers, and indeed the use of the network today is more intelligent than in the first months of LHC running where we observed some extremely large data transfer rates.

That work has continued through specific projects between the experiments together with the WLCG development teams; and additional work and discussions during the TEG activity in 2012. In some cases, such as the work of federated data mechanisms using xrootd, this has been a common effort between several experiments.

Some of the main storage and data management strategies currently being followed are described in the following sections.

8.2.1 Distinguish data archives from disk pools

Over the past generation of experiments the tape archive has become more and more integrated with the disk pools from which the user accesses the data, with automated migration of data to and from tape. However, there is a huge disparity in data bandwidth between the data consumer and the tape disk systems. There is also a significant latency in accessing data on tape in a robot. The software that manages such integrated HSM systems is complex and costly to support. Large-scale tape accesses by random users can also bring the tape service to a halt if not carefully controlled. For this reason the LHC experiments have largely prevented access to tape directly by users other than the production managers. Also because of the scale of the data sets it is important that the recall from tape of significant data sets be well organised and managed – together with the disk space needed to hold that recalled data. Thus the production teams explicitly manage data recall from tape, and the use case for automated recall disappears. It has also been noted that often it is faster to access a remote disk copy of a file than to recall it from the local tape system.

This change actually simplifies the data storage model for the large archive sites, and allows a factorisation of the management software (but does not force this), which helps to reduce complexity. This also allows a decoupling of the technology used for the archive and the disk pools, and potentially better control of disk management at Tier 1 sites. Ultimately this can facilitate a separation of the archiving functionality.

No additional tools are necessary to implement this separation, as the FTS tool can be used.

As noted above, for the large experiments it is important to limit the number of tape accesses. This is already the case for ATLAS, while CMS had assumed that

tape would be an active part of the system in their original computing model but is now also moving in this direction. This potentially changes the underlying assumptions on the amount of disk space necessary however. LHCb and ALICE also explicitly manage tape access.

8.2.2 The use of SRM

SRM was conceived as a mechanism through which access to often very different mass storage systems could be managed without the user job needing to know the details of different APIs and tools. However, the implementations of SRM often caused performance bottlenecks and added complexity to the system. With the evolution of data storage with 2 distinct changes: the separation of archive and disk noted above, and the move towards common interface to disk systems (such as xrootd or http), the role of SRM has become less important.

Thus the expectation is that SRM will remain as the interface to archives and managed storage, albeit with a well delineated (sub-)set of the SRM functionality. That definition was done by the TEG work, and thus SRM is no longer a bottleneck to the mass storage services. Clearly SRM is simply not needed (and often not implemented) as the interface to disk-only storage services.

The new version of the File Transfer Service (FTS-3) does not require SRM, as it can talk to gridftp directly as well as other interfaces).

Thus, there is no requirement to replace SRM explicitly as an interface, but there is no longer any need for it in the future.

Finally, as future interfaces to storage, there is interest in understanding how cloud storage will evolve, but this is not at all well defined today. This technology will be tracked.

8.2.3 Data security models

It is important to recognise that fine-grained authorisation is a process with a relatively high latency and processing overhead due to the nature of the X509 certificate and infrastructure. In particular for analysis jobs that may access many hundreds of files, the overhead becomes important. Thus it is important that authorisation be done at the appropriate level and granularity. For example read-only cached data may have a simpler security model and ovoid the overheads.

The security working group reported on work that is needed to ensure that data and storage services have adequate protection against error and misuse, while optimising the overheads involved. There are also related issues of data ownership that need to be addressed.

8.2.4 Stateless data services for smaller sites

As discussed earlier for sites where there is limited support effort available, or for using opportunistic resources where installing special grid software may be problematic, there is a desire to move towards and automated configuration of the site, with a limited amount of set up required.

Thus, it would be advantageous to be able to distinguish between (Tier 2) sites that must provide permanent data storage, and those that are merely caches for

working data. The latter type of sites may have lower levels of service. Data cache functionality could be provided through simple storage services such as squids (if http as a protocol is used), or a simple file system.

One of the points noted by the review of operations over recent years is the importance of having robust storage services. Again this would lead in the direction of smaller sites with less support effort, being better configured as stateless caches, and not requiring complex storage management software to be installed and managed by an expert.

8.2.5 Data federation and remote data access

Building data federations with xrootd is a clear direction, already proposed in 2010 following the Amsterdam workshop. Both ATLAS and CMS are currently implementing this for certain use cases, and it has already been in widespread use by ALICE for several years. Care over efficiency is important to avoid having too many jobs waiting on remote data. Thus monitoring of the use of such remote I/O is important in optimising the overall system efficiency.

There are several specific use cases where using data remotely is helps improve usability and efficiency of the system:

- First is the fall back channel for failed file opens, for example when a data set has a large number of files, and one fails, accessing that file remotely, rather than the job failing;
- Can be used also to force applications to run on non-resident data for example for debugging and visualisation;
- Small diskless Tier-3 sites (or opportunistic sites that don't have local storage). This has been the model for several cloud use cases;

In the longer term the data federation may allow a more optimised use of storage – for example fetching a missing file from another site rather than falling back to tape. Automation of this may lead to missing data being fetched transparently when necessary so that applications may not need to concern themselves over the complete availability of all the files of a dataset.

8.2.6 Data popularity and intelligent data placement (Maria)

Need a summary of what is happening

8.2.7 Data transfer service and protocols

The WLCG data transfer services have been one of the keys to the success of the management of the huge volumes of LHC data, and bringing those services and underlying infrastructure up to scale was the subject of many of the large scale data challenges prior to the LHC switch on. So far, all of that bulk data movement has relied on the gridftp protocol, which originated as part of the original Globus grid implementation. While successful, this relies on dedicated software that must be maintained by the grid community. Today, there are many examples of large-scale data movement using more common protocols such as http. That also has the advantage that there are many off-the-shelf solutions to storage and data management using http (such as web caches, redirection for load balancing or fail-over, etc.), that would not need dedicated support from the HEP community.

There has been work invested (e.g. by the DPM and the dCache teams) in implementing http as an alternative to gridftp. While there is no urgency to move away from gridftp, there is a general recognition that long-term sustainaibility may be better addressed by moving to such standard protocols and tools. For example the data federation currently being implemented with xrootd could well eventually be done with http and web services.

A side benefit of such an eventual move would be the better attractiveness of WLCG tools to other non-HEP communities.

8.2.7.1 File Transfer Service (FTS-3)

The on-going work on the latest version of the File Transfer Service (FTS-3) is recognised as a high priority. This version is a major re-write of the software, taking into account all the lessons of the experience with LHC data. This version could also be centrally deployed for all of WLCG if required (previous versions had to be deployed at all Tier 1s). It also supports network mesh model (no strict hierarchy between Tiers), and is far more flexible than before and simpler to configure than before. It is foreseen that FTS-3 could also be the tool used to manage the organised data recall from tape. As noted previously, FTS-3 does not require an SRM endpoint.

8.2.8 Storage accounting

Storage accounting is a functionality that has been missing, and has been requested several times. However, it is not as straightforward as CPU-time accounting, since the usage of storage systems must be accounted at a fixed point in time (whereas CPU accounting is simply a cumulative total over a given period), and secondly disk systems tend to be close to full most of the time, particularly if they are operated as caches. Thus strict space accounting is not such a useful measure.

However, there is a proposal from the EMI[iii] project for a standard accounting record. This proposal has been accepted and will be used as the basis for reporting on the use of disk storage systems. Such an implementation requires that each storage service implementation provide the data needed in this record. This work is on going, and there will be an extension to the CPU accounting portal, to report on disk space utilisation.

What is perhaps more interesting is to understand how much of the data on a given storage system is active, how often it is used, and how long it stays resident on the storage system. This information is available from the data popularity service discussed above, and currently being implemented for the experiments.

8.2.9 I/O Classification and Benchmarking Working Group

The goal of this working group is to provide a set of easy-to-use benchmarks that simulate storage access of a realistic ensemble of WLCG user jobs for storage optimization and access model comparisons. The methodology chosen is to analyse measured access patterns – now available from storage system logs and federation wide log collectors – in order to classify the dominant access patterns and their respective contributions as statistical distributions in a set of key metrics. After first sanity checks, to insure that the measured behaviour is

indeed intended and effective, these (parameterized) distributions are then used to provide a standalone load generator, which reproduces user behaviour without the need of installing a larger body of experiment framework or user code.

In the initial phase of this project, data from several experiments has been analysed and a first set of classification metrics has been proposed. The working group has discussed in several meetings apparent differences between the experiment use patterns - for example in the distribution of achieved transfer speeds and in the rate of reopening input files - to understand the origin of those differences. A suite of micro benchmarks is now being constructed, which based on the measured input distributions recreates matching I/O patterns. At this point the package still needs confirmation and tuning. Once additional log information about the fraction of vector-reads becomes available from more recent storage back-ends, this can be integrated to estimate the impact of WAN network latencies on aggregate access performance in a federated storage environment.

8.3 Database services

There are several database services in use by the experiments for file catalogues, conditions databases and other uses. However, the complexity of some of the database deployments and usage models have been underlying some of the service incidents related to the Oracle database services. Thus reducing complexity and simplifying demands on the database services will also improve robustness of the overall system and reduce the operations load.

ATLAS, CMS, and LHCb all use Oracle databases for their conditions data, and ATLAS and LHCb use the COOL implementation (a common project developed by LCG), while CMS has its own solution. All three experiments use CORAL as the layer that connects to different back-end technologies. There is no plan to significantly revisit the conditions database model and tools, although there is work to improve performance by reducing database accesses by jobs. The COOL and CORAL services will continue to be supported. However, the earlier model of Oracle database replication has now been replaced by providing the conditions data through CVMFS, Frontier, and other caching mechanisms. This in itself has simplified the Oracle deployment to essentially a central service.

Other areas where replication of Oracle databases is still important have moved to the newer Oracle Active Data Guard in place of the original Streams technology, and the use cases are generally simpler.

ATLAS and LHCb have used the LFC as a file catalogue service. Originally ATLAS had a very complex LFC deployment with instances for each of their Tier 1 clouds. This has now been consolidated to a central LFC with replication to BNL for backup.

It is anticipated that ATLAS and LHCb will eventually migrate away from the LFC.

As well as the traditional relational database services, there have recently been lots of diverse experimental uses of Hadoop and NoSQL tools; with many potential use cases.

In order to facilitate the understanding of these tools and the needs for any eventual services, CERN is providing a pilot Hadoop service. The use and support of the (potentially very many) higher-level tools is seen as the responsibility of the application at the moment.

8.4 Operations and Infrastructure Services

There has been a WLCG operations coordination activity in place since the first data challenges in 2004. This has worked closely with the grid infrastructure operations activities in the EGEE, EGI, and OSG projects. In reviewing the needs for the future, it was recognised that some consolidation of various operations-level activities would be beneficial, and the WLCG service and operations coordination team was proposed to implement that consolidation. This team was put in place in 2012, and is the core team managing a set of related functions:

- Managing and ensuring follow-up of all operational issues;
- Coordination of integration activities – such as the ensuring new or updated services are correctly integrated with the experiments' computing services;
- Managing the deployment and testing of new or updated services, using the staged-rollout mechanism to selected sites;
- Coordination of operations across WLCG, EGI, OSG and others as appropriate.

This activity is very much a collaborative one, with effort drawn from across the WLCG collaboration. It has also consolidated the various different operation-related meetings.

This core team may need to eventually absorb some of the tasks currently delegated to EGI, although this is very much dependent on how the EGI operation will evolve following the end of the current project funding round in 2014.

8.4.1 Computing as a Service

A very strong feedback from the WLCG sites during the review of operations was the expression of the need to move towards a model described as "Computing as a Service", particularly at smaller sites. This need has been expressed several times in the present document for a variety of reasons, but from the operational point of view the need is to simplify the software and middleware that must be deployed, in terms of both functionality and deployment. As noted previously such a move will help to get to a situation where smaller sites can be stateless, close to zero configuration, and minimal operational effort required, as well as helping to easily leverage opportunistic resources.

8.4.2 The software lifecycle model

Following the end of the EMI middleware project, there was a need to clarify the needs from WLCG for the on-going management of grid middleware which had until 2013 been supported by EGEE and then by EMI for the European

middleware. At the same time, OSG has also reviewed its middleware support mechanisms. Today the models and tools to be used are similar and follow similar rationales. Some of the key points are the following:

- Most grid middleware is now stored in the EPEL (Redhat) open source repository, which imposes a certain methodology and process for packaging;
- Additional software that is not suitable to be put into EPEL is stored in a dedicated WLCG repository, which may be managed by EGI, for example; there are other independent repositories (e.g. for dCache);
- The build systems are now based on standard open source tools.

Consequently the majority of grid middleware is now in a more usual open-source environment, and opens the possibility for more collaborative maintenance and development. This feature is important for the long term sustainability of the infrastructure.

There is still a need to ensure that appropriate testing is carried out. However the responsibility for basic testing lies with the developers of each package or service, and the larger scale testing requires the organised staged-rollout mentioned above.

8.5 Security Aspects

In the review of security performed in 2012, a risk analysis was undertaken (updating that originally performed several years ago). The summary of that analysis is given here. Eleven distinct risks were identified and ranked. The ranking of a risk is quantified as the product of the likelihood of an event and its probably impact.

1. High Risk:
    a. Misused identities (including "ssh" identities, not only grid credentials)
    b. Attack propagation between WLCG sites
    c. Exploitation of a serious operating system vulnerability
2. Medium Risk:
    a. Threats originating from trust services
    b. Negative publicity on a non-event
    c. Insecure configuration leading to undesirable access
    d. Insufficient protection of information leading to sensitive data leakage
3. Low Risk:
    a. Incidents of resources not bound by WLCG policies
    b. Exploitation of a serious application or middleware software vulnerability
    c. Data removal, corruption, or alteration
    d. Denial of Service attack from an external organisation

This analysis highlighted the need for fine-grained traceability of which user credential was connected to a given action. This is essential in order to contain and investigate incidents, and to try and prevent recurrence. This requirement on traceability has always been part of the security policy, but with the use of

pilot jobs the connection between running workload and the submitting user is no longer direct. The deployment of the glexec service is required to address this in the short term.

The WLCG security team, together with the security activities in the grid infrastructures (EGI, OSG) are collaborating to continue to identify areas where risks must be managed, to prioritise where efforts to improve security should be directed, and to assess the cost versus benefits of particular measures.

Areas where on-going work is required include:

- Fulfilling the traceability requirement by fully deploying the glexec service at all grid sites, and ensuring that it is used by the experiment pilot frameworks;
- Enabling appropriate security controls, such as the centralised flagging and eventual banning of compromised user credentials, or misbehaving users;
- Training and education to ensure that appropriate and best-practice security practices exist at all collaborating sites (such as regular OS patching, monitoring, etc.);
- Collaborating with external communities on incident response and policy development;
- Maintaining the existing networks of trust (at policy and operational levels).

8.6 Distributed Computing Services (middleware)

The set of services required

- ARGUS, VOMS, glexec,

8.7 Managing the future evolution of the infrastructure

8.8 Data Preservation and Open Access Infrastructure

<Input (1 page) from Jamie>

8.8.1 ALICE

The data harvested by the ALICE Experiment up to now and to be harvested in the future constitute the return of investment in human and financial resources by the international community. These data embed unique scientific information for the in depth understanding of the profound nature and origin of matter. Because of their uniqueness, long-term preservation must be a compulsory objective of the data processing framework and will lay the foundations of the ALICE Collaboration legacy to the scientific community as well as to the general public. These considerations call for a detailed assessment of the ALICE data preservation strategy and policy. Documentation, long term preservation at various levels of abstraction, data access and analysis policy and software availability constitute the key components of such a data preservation strategy allowing future collaborators, the wider scientific community and the general public to analyse data for educational purpose and for eventual reassessment of the published results.

To reinforce these messages and to put them in practice, the ALICE Collaboration is in process of agreeing on Data Preservation Strategy document which will be in line with similar documents already drafted by some other experiments and stipulate the principle of open access to the data, software and documentation, as well as the processing of data by non-ALICE members under the conditions listed in the policy implementation document.

8.8.2 ATLAS

It is an increasing obligation of all science projects to put into effect a data management and access policy that extends beyond the confines and lifetime of the experiment. ATLAS has an expected lifetime of decades, but effective long term preservation requires action be taken early in the design of the data objects and software, such that it will be reusable long after the last collision is recorded. Furthermore, due consideration and effort must be made so that at least some form of the data can, after a reasonable embargo period, be used by non-collaboration members. ATLAS is currently evolving a policy on these matters. The policy must inform the other software and computing efforts, and will require effort over and above the normal exploitation of the data.

8.8.3 CMS

CMS has approved a data preservation, re-use and open access policy[9], which motivates and defines the CMS approach to the preservation of the data and access to them at various levels of complexity. The implementation of the policy has been elevated to a dedicated project within the collaboration, covering areas from open access at different levels and, analysis knowledge and bit-level preservation. CMS is looking for solutions, which could be usable for the other LHC experiments, and promotes common infrastructures wherever possible.

The CMS data policy emphasizes the importance of the possibility of re-use of the CMS data. In parallel to the open access papers, publication of additional numerical data in form in which they can be easily extracted for further use is encouraged, as well as initiatives such as RIVET[10] [2] allowing for easy comparison between observed data and Monte Carlo event generators and their validation. For the distribution of these data, CMS relies on digital library services such as INSPIRE and HEPData.

Going to the level of reconstructed data and its re-use, the experience from the other HEP experiments indicates that the biggest challenge in data preservation is the loss of knowledge and expertise. While CMS, and HEP experiments in general, do very well in recording "immediate" metadata, such as event and run numbers, beam conditions, software versions used in the data reprocessing, we are doing poorly on "context" metadata, i.e. practical information needed to put the data in context and analyze them. Therefore, CMS is actively looking for solutions to enable easy recording this detailed knowledge, readily available at

[9] CMS Collaboration, “CMS data preservation, re-use and open access policy”:

https://cms-docdb.cern.ch/cgi-bin/PublicDocDB/ShowDocument?docid=6032

[10] http://rivet.hepforge.org/

the time of the active physics analysis, but quickly forgotten. The Invenio team at CERN, with the input from CMS, will setup a prototype of a tool, which could make recording, and documenting of the relevant details easy. This documentation could be available to the collaboration members only, and only the part of it, which is considered appropriate, could go to public domain.

CMS is currently preparing for a public release of part of 2010 collision and simulated data in AOD format. We expect that regular public releases will be made of the data after it has been analyzed by the collaboration. To make things easier for open access users, a virtual machine (VM) image has been prepared, in a format usable by the freely available VirtualBox application. For the access to the data, the initial workflow is kept as close to the standard one as possible, which uses xroot. An xrootd server has been commissioned with anonymous read-only access, further limited by fi?rewall to include only those sites involved in the testing phase.

A framework to enable long-term validation of CMS data is being prepared. CMS has powerful tools for validation of software releases and for data quality monitoring. The goal of the long-term validation project is to extract and record the necessary information and tools in order to be able validate the data and software in case of re-use in long-term future.

CMS works in collaboration with the CERN-IT department to define different tools and services covering the areas and levels of data preservation mentioned above. Discussions on such a service provision are ongoing.

At the bit-level, while CMS computing model offers a solid base for long-term data preservation, CMS is looking forward to the program of work of the bit-level data preservation working group under the HEPiX forum, where the challenges in this area will be discussed in the context of WLCG.

8.8.4 LHCb

LHCb has adopted a policy on data access in 2013. In this LHCb have affirmed their wish to deal with both data preservation, and building on that work, data access. In the first instance LHCb is working towards ensuring that all data can be re-accessed at any time in the future in order to check an analysis or carry out a new analysis.

This analysis may include re-reconstructing the RAW data or may use the latest version of the DST data - for example to add a new stripping line. The LHCb performance and regression-testing framework will be used to ensure that new releases of the code can successfully read old data.

Data recovery procedures will be developed to protect against data corruption and losses. In addition work is ongoing to ensure that MC data sets corresponding to old conditions can be generated. Once this work is complete it is the intention to use this infrastructure to make subsets of the data available, at some point after it was recorded according, to the policy in force.

[i] http://isscc.org/doc/2013/2013_Trends.pdf

[ii] Technical Evolution Group reports: http://cern.ch/lcg/news/teg-reports

[iii] European Middleware Initiative; EC-funded project: http://www.eu-emi.eu