

CERN-LHCC-2005-018
ALICE TDR 012
15 June 2005

ALICE

Technical Design Report

of the

Computing

Cover design by CERN Desktop Publishing Service.
Visualisation of ALICE Experiment as simulated by AliRoot.

Printed at CERN
June 2005.

ISBN 92-9083-247-9

ALICE Collaboration

Contents

3	Distributed computing and the Grid	3
3.1	Introduction	3
3.2	Distributed computing	3
3.3	AliEn, the ALICE interface to the Grid	5
3.4	Future of the Grid in ALICE	8
	References	11

3 Distributed computing and the Grid

3.1 Introduction

The ALICE computing model assumes the existence of a functional Grid middleware allowing for efficient, seamless, and democratic access to worldwide-distributed heterogeneous computing and storage resources. The technical feasibility of this assumption has been demonstrated during the series of Physics Data Challenges (PDCs) in 2003 and 2004 with the ALICE-developed AliEn [1] system.

Our current plan is to build a similar system making maximum usage of the common Grid services deployed by LCG [2] and adding the necessary ALICE-specific services derived from the AliEn system.

The experience gained during the data challenges strongly supports a Grid model with minimum of intrinsic hierarchy and specific resources categories. Intelligent workload scheduling based on the advertised features of the computing resources and on the job requirements allows for a better utilisation of the resources with respect to a fixed hierarchy. However, for clarity and to account for the possibility to have less advanced Grid services, the foreseen computational tasks and classes are categorized below in a manner that caters to a more commonly accepted ‘tiered’ distributed computing centre architecture.

The tier structure allows for a layered classification whereby, depending on the type of computational tasks, tasks are assigned to a computing element with a specific functionality. However, it also introduces a more rigid access to the resources, requiring a higher level of central management and thus increases the overall cost of the final system in terms of CPU, storage and manpower.

In this chapter a short description of the ALICE distributed computing environment is presented, with emphasis on the ALICE view of a decentralized Grid structure with advanced functionality. Owing to the rapid evolution of the Grid, it is expected that the distributed computing environment will undergo several modifications, especially in the area of the Grid implementation details.

3.2 Distributed computing

The ALICE computing model is driven by the consideration of the large amount of computing resources which will be necessary to store and process the data generated by the experiment. Detailed description of the data sizes under different data-taking scenarios are given in Chapter ??.

Since the conceptual design of the LHC [3] experimental programme, it was recognized that the required data processing and storage resources cannot be consolidated at a single computing centre. It is more natural, considering both the substantial financial investment involved and the need for expert human resources, that these resources are distributed at the HEP computing facilities of the institutes and universities participating in the experiment. The technical side of the decentralized offline computing scenario has been formalized in the so-called MONARC model [4] schematically shown in Fig. 3.1. MONARC describes an assembly of distributed computing resources, concentrated in a hierarchy of centres called Tiers, where Tier 0 is CERN, Tier 1s are the major computing centres which provide a safe data storage, likely in the form of a mass storage system (MSS), Tier 2s are smaller regional computing centres. The MONARC model also foresees Tier 3s which are university departmental computing centres and Tier 4s that are user workstations; however the distinction of these lower Tiers is not relevant in a Grid model. The major difference between the first three Tiers is the quality of service (QoS) and reliability of the computing resources at every level, where the highest QoS is offered by the Tier 0/Tier 1 centres. At the moment of publication of this document, the QoS metrics associated to the different Tiers are still under discussion.

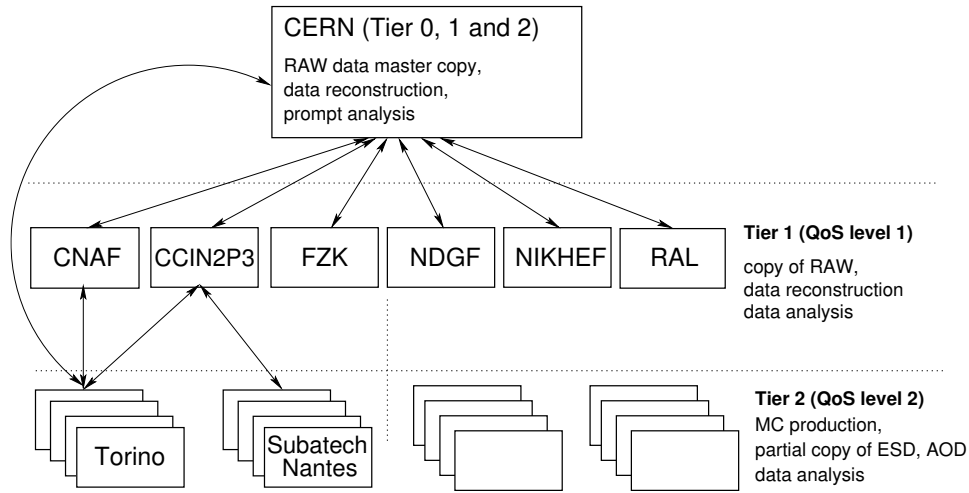


Figure 3.1: Schematic view of the ALICE offline computing tasks in the framework of the tiered MONARC model. The acronyms are explained in the text.

The basic principle underlying the ALICE computing model is that every physicist should have equal access to the data and the computing resources necessary for its processing and analysis. Thus, the resulting system will be very complex with hundreds of components at each site with several tens of sites. A large number of tasks will have to be performed in parallel, some of them following an ordered schedule, for example the raw data reconstruction, large Monte Carlo production, data filtering and stripping, and some being largely unpredictable: single-user Monte Carlo production and data analysis. In order to be used efficiently, the distributed computing and storage resources will have to be transparent to the end user, essentially looking like a single entity.

The commonality of distributed resources management is being realized under the currently ongoing development of the Grid [5]. It was conceived to facilitate the development of new applications based on high-speed coupling of people, computers, databases, instruments, and other computing resources by allowing “dependable, consistent, pervasive access to high-end resources”. Although the MONARC model pre-dates the appearance of the Grid concept, its terminology is well adapted to the distribution of resources that are present in the HEP community and remains very useful for discussing the organization and relations of the centres. In this document, we shall consider only Tier 0, Tier 1 and Tier 2 functionality.

However, in a well functioning Grid, the distribution of tasks to the various computing centres will be performed dynamically, based on the availability of resources and the services that they advertise, irrespective of the assigned Tier level. This picture is an evolution of the MONARC structure, since it is more flexible and allows for a better optimisation of resource usage. This Tier-free model is sometimes called ‘Cloud Model’ and it has been adopted as a concept in the development of the AliEn system and used to exploit the computing capacities offered by the computing centres participating in the ALICE Grid during the Physics Data Challenges.

The ALICE computing model foresees that one copy of the raw data from the experiment will be stored at CERN (Tier 0) and a second copy will be distributed among the external (i.e. not at CERN) Tier 1 centres, thus providing a natural backup. Reconstruction to the Event Summary Data (ESD) level will be shared by the Tier 1 centres, with the CERN Tier 0 responsible for the first reconstruction pass. Subsequent data reduction to the Analysis Object Data (AOD) level, analysis and Monte Carlo production will be a collective operation where all Tier 1 and 2 centres will participate. The Tier 1s will perform reconstruction and scheduled analysis, while the Tier 2s will perform Monte Carlo and end-user analysis.

Grid technology holds the promise of greatly facilitating the exploitation of LHC data for physics

research and ALICE is very active on the different Grid test-beds and worldwide projects [2], [6], where Grid middleware prototypes are deployed. The objective of this activity is to gain experience with all systems, representing a reasonable amount of computing resources, and ultimately assemble a computing environment which is able to fulfil the collaboration needs in terms of offline data production and analysis.

This activity, both very useful and interesting in itself, is faced with the different levels of maturity and functionality of the deployed middleware and its various flavours emerging from different Grid projects and communities. Any middleware currently available is largely a result of leading-edge computer science research and is therefore still rather far from production quality. Moreover, there are no commonly adopted inter-Grid communication standards and the development in different projects is following very different paths and software standards, even if the functionality which is aimed at is very similar.

The emerging heterogeneous picture is rather unfavourable, since the computing resources to be exploited by the experiments like ALICE are presented in the form of non-conformal Grid implementations. In addition, owing to the multitude of projects working independently on a solution to essentially the same problems, the most complex tasks to be executed on the Grid, e.g. distributed data analysis, cannot be adequately addressed.

Faced with this situation, the LHC experiments are developing a very similar approach, although on parallel lines. They are planning to deploy services offering a homogeneous view of the Virtual Organisation corresponding to the experiment, while these services are in turn interfaced to the different Grid flavours deployed on the resources offered to the experiment by the funding agencies. In a non-dissimilar way, ALICE is planning to use some of the AliEn services to interface with the different Grids. It is of course in the best of ALICE's interest to minimise the amount of experiment specific services that we will have to deploy and maintain ourselves, maximising the use of the common Grid services deployed and maintained by the computing centres.

3.3 AliEn, the ALICE interface to the Grid

The AliEn (**Ali**CE **En**vironment) framework has been developed with the aim of offering to the ALICE user community a transparent access to computing resources distributed worldwide through a single interface. During the years 2001–2004 AliEn has provided a functional computing environment fulfilling the needs of the experiment in its preparation phase. AliEn was primarily conceived as the ALICE user entry point into the Grid world, shielding the users from its underlying complexity and heterogeneity. Through interfaces, it can use transparently resources of different Grids developed and deployed by other groups. This advanced concept has been successfully demonstrated during the ALICE Physics Data Challenge '04 (PDC'04), where the resources of the LCG and INFN Grids were accessed through interfaces. Approximately 11% of the computing resources used in PDC'04 were provided by the LCG and INFN Grids. In the future, the cross-Grid functionality will be extended to cover other Grid flavours. In addition, AliEn has been engineered to be highly modular, and individual components can be deployed and used in a foreign Grid, which is not adapted to the specific computational needs of ALICE, thus achieving an efficient use of the available resources.

The system is built around Open Source components and uses a Web Services model [7] and standard network protocols. Less than 5% is native AliEn code (mostly code in PERL), while the rest of the code has been imported in the form of Open Source packages and modules.

Web Services play the central role in enabling AliEn as a distributed computing environment. The user interacts with them by exchanging SOAP (Simple Object Access Protocol) messages and they constantly exchange messages between themselves behaving like a true Web of collaborating services. AliEn consists of the following components and services: authentication, authorization and auditing services; workload and data management systems; file and metadata catalogues; the information service; Grid and

job monitoring services; storage and computing elements. A schematic view of the AliEn services, their location and interaction with the native services at the computing centres is presented in Fig. 3.2.

The AliEn workload management system is based on the so-called ‘pull’ approach. A service manages a common task queue, which holds all the jobs of the ALICE Virtual Organization (VO). On each site providing resources for the ALICE VO, Computing Element (CE) services act as ‘remote queues’ giving access to computational resources that can range from a single machine, dedicated to running a specific task, to a cluster of computers in a computing centre, or even an entire foreign Grid. When jobs are submitted, they are sent to the central queue. The workload manager optimizes the queue taking into account job requirements such as the input files needed, the CPU time and the architecture requested, the disk space request and the user and group quotas. It then makes jobs eligible to run on one or more computing elements. The CEs of the active nodes get jobs from the central queue and deliver them to the remote queues to start their execution. The queue system monitors the job progress and has access to the standard output and standard error.

Input and output associated with any job are registered in the AliEn File Catalogue (FC), a virtual file system in which logical names, with a semantics similar to the Unix file system, are assigned to files. Unlike real file systems, the FC does not own the files; it only keeps an association between one or possibly more Logical File Names (LFN) and (possibly more than one) Physical File Names (PFN) on a real file or mass storage system. The correspondance is kept via the **G**lobal **U**nique file **I**dentifier (GUID) stored in the FC. The FC supports file replication and caching and it provides the information about file location to the RB when it comes to scheduling jobs for execution. These features are of particular importance, since similar types of data will be stored at many different locations and the necessary data replication is assumed to be provided transparently and automatically by the Grid middleware. The AliEn file system associates metadata with LFNs.

ALICE has used the system for distributed production of Monte Carlo data, reconstruction and analysis at over 30 sites on four continents. The round of simulation, reconstruction and analysis during the PDC’04 was aimed at providing large amounts of simulated data for physics studies as well as testing the main components of the ALICE computing model. During the data challenge, more than 400 000 jobs were successfully run worldwide from the AliEn Task Queue (TQ), producing 40 TB of data. Computing and storage resources were available both in Europe and the US. The amount of processing needed for

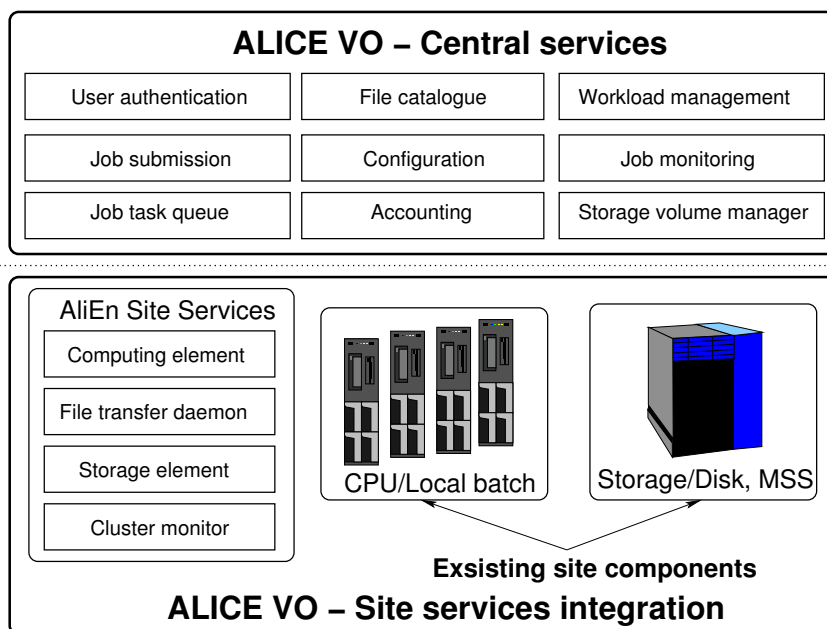


Figure 3.2: Schematic view of the AliEn basic components and deployment principles.

a typical production is in excess of 30 MSI2k×s to simulate and digitize a central Pb–Pb event. Some 100 000 high-multiplicity Pb–Pb events were generated for each major production. This is an average over a very large range since peripheral events may require one order of magnitude less CPU, and pp events two orders of magnitude less. The Pb–Pb events are then reprocessed several times superimposing known signals, in order to be reconstructed and analysed. Again there is a wide spread in the time this takes, depending on the event, but for a central event this needs a few MSI2k×s. Each Pb–Pb central event occupies about 2 GB of disk space, while pp events are two orders of magnitude smaller. The total amount of CPU work during PDC'04 was 750 MSI2k×h. The relative contribution of the computing centres, participating in the ALICE Grid during PDC'04 is shown in Fig. 3.3.

The Grid user data analysis has been tested in a limited scope using tools developed in the context of the ARDA project [8] (the gShell interface to the FC and the analysis tools based on it). Two approaches were prototyped and demonstrated: the asynchronous (interactive batch approach) and the synchronous (true interactive) analysis.

The asynchronous model has been realized by extending the ROOT [9] functionality to make it Grid-aware. As the first step, the analysis framework has to extract a subset of the datasets from the file catalogue using metadata conditions provided by the user. The next part is the splitting of the tasks according to the location of datasets. Once the distribution is decided, the analysis framework splits the job into sub-jobs and inserts them in the AliEn TQ with precise job descriptions. These are submitted to the local CEs for execution. Upon completion, the results from all sub-jobs are collected, merged and delivered to the user. This model has been shown to work with satisfactory results, but more attention has to be devoted to the optimisation of the load on the FC, when many simultaneous large user queries are performed and the fault-tolerance of the merging mechanism when some of the sub-jobs fail.

The synchronous analysis model requires a much tighter integration between ROOT and the Grid services, where the framework should be able to execute in parallel and in real-time all sub-jobs associated to the main user job. In addition, the system should automatically scale the number of running processes to the amount of available resources at the time of execution. The model relies on extending

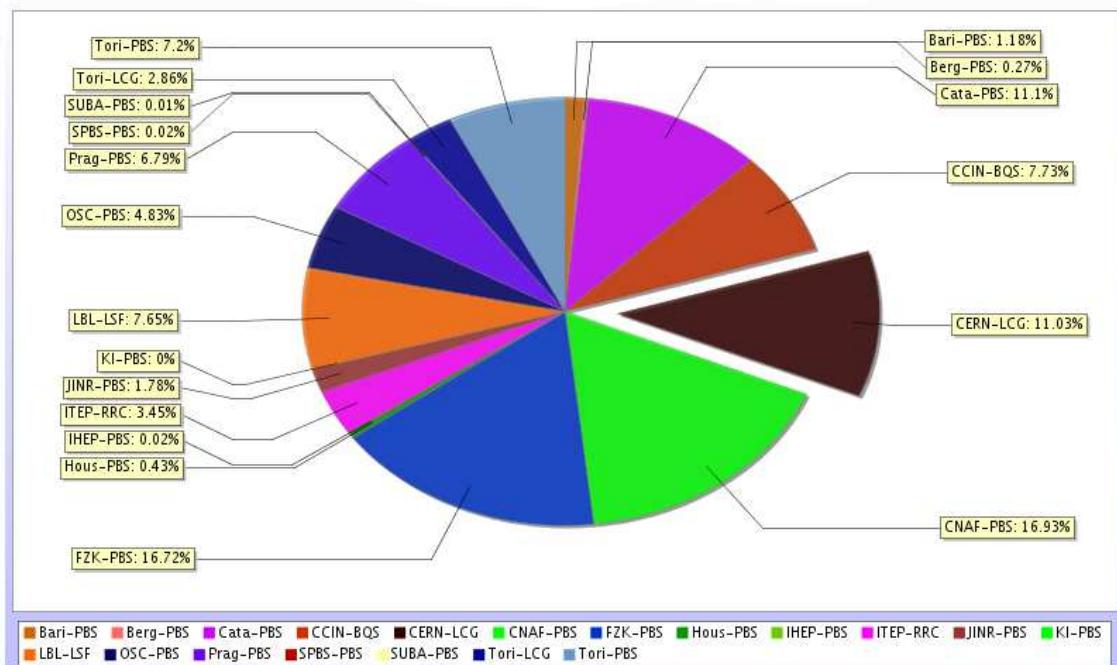


Figure 3.3: Relative CPU and storage contribution by the participating sites during the PDC'04.

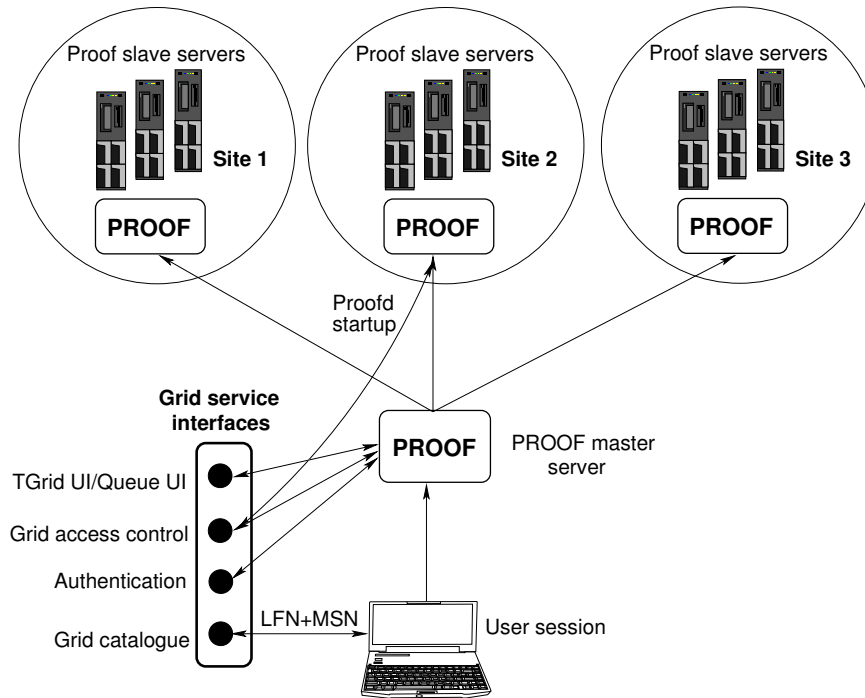


Figure 3.4: Setup and interaction with the Grid middleware of a user PROOF session distributed over many computing centres.

the functionality of PROOF [10] – the Parallel ROOT Facility. The PROOF interface to Grid-like services is presently being developed, focusing on authentication and the use of the FCs, in order to make both accessible from the ROOT shell.

In the conventional, single-site setup, PROOF workers are managed by a PROOF master server, which distributes tasks and collects results. In a multi-site setup, each site running a PROOF environment will be seen as a PROOF worker for a PROOF master running on the user machine. The PROOF master has therefore to implement the functionality of a master and a worker at the same time. This concept is illustrated in Fig. 3.4. AliEn classes used for asynchronous analysis as described earlier can be used for task splitting in order to provide the input data sets for each site that runs PROOF locally.

The AliEn-PROOF-based system for distributed synchronous analysis will be used for a rapid evaluation of large data samples in a time-constrained situation, for example the evaluation of the detector calibration and alignment at the beginning of a data-taking period. This will allow for an efficient planning of critical analysis tasks, where the predictability of the execution time is very important. As such it is an essential building block of the ALICE computing model.

3.4 Future of the Grid in ALICE

The experience with the AliEn Grid middleware has been instrumental in shaping the ALICE computing model. The technical feasibility of a functional Grid, effectively managing thousands of processors and hundreds of terabytes of storage, distributed over many computing centres worldwide has been demonstrated in a series of realistic Physics Data Challenges. These were conceived and executed with parameters closely approximating the real running conditions of the ALICE experiment.

One of the most important requirements for the efficient use of the Grid is the existence of a single interface to the computing resources, effectively shielding the end user from the underlying software and hardware complexity. In the past several years this role has been played by AliEn acting as a complete vertical Grid system.

During this time, different Grid solutions developed by large collaborations, have come to maturity. However, as explained before, two major issues still remain. First of all, none of these Grid flavours provides a complete solution for the ALICE computing model, and secondly there are no accepted standards, and all these Grids provide a different user interface and a diverse spectrum of functionality.

Therefore some of the AliEn services will continue to be used as the ALICE's single point of entry to the computing resources encapsulated by the various Grid entities and as a complement of their functionality to implement the ALICE computing model. In this model, which has already been prototyped and used, the foreign Grid will be accessed via interfaces. The elements of AliEn used in synergy with the deployed Grid middleware to supplement its functionality will assure efficient use of the computing resources. The cross-Grid role of AliEn preserves the present ALICE infrastructure and user access methods and allows for its continuous development and enrichment with advanced functionalities. It also provides the necessary methods for addition of computing resources and adoption of new Grid standards as they become available.

The ALICE Computing Project will closely watch the evolution of the functionality of the deployed Grid middleware and of the international Grid standards. Whenever a standard Grid service is found to provide the same, or better, functionality than an ALICE-specific one, its adoption will be considered in order to reduce the maintenance load and increase portability and robustness of the ALICE computing environment.

References

Chapter 3

- [1] P. Saiz *et al.*, Nucl. Instrum. Methods **A502** (2003) 437-440;
<http://alien.cern.ch/>.
- [2] <http://lcg.web.cern.ch/LCG/>.
- [3] <http://lhc-new-homepage.web.cern.ch/lhc-new-homepage/>.
- [4] <http://www.cern.ch/MONARC/>.
- [5] The Grid: Blueprint for a New Computing Infrastructure I. Foster, C. Kesselmann (Eds), M. Kaufmann, 1999;
I. Foster, C. Kesselman, S. Tuecke, The Anatomy of the Grid Enabling Scalable Virtual Organizations, see, <http://www.globus.org/research/papers/anatomy.pdf>.
- [6] <http://egee-intranet.web.cern.ch/egee-intranet/gateway.html>.
- [7] <http://www.w3.org/2002/ws/>.
- [8] <http://lcg.web.cern.ch/LCG/peb/arda/Default.htm>.
- [9] <http://root.cern.ch/>.
- [10] M. Ballintijn, R. Brun, F. Rademakers and G. Roland, *Distributed Parallel Analysis Framework with PROOF*, Proc. of TUCT004.

