A science-gateway workload archive to study pilot jobs, user activity, bag of tasks, task sub-steps, and workflow executions

Rafael Ferreira da Silva and Tristan Glatard

University of Lyon, CNRS, INSERM, CREATIS, Villeurbanne, France
{rafael.silva,glatard}@creatis.insa-lyon.fr

Abstract. Archives of distributed workloads acquired at the infrastructure level reputably lack information about users and application-level middleware. Science gateways provide consistent access points to the infrastructure, and therefore are an interesting information source to cope with this issue. In this paper, we describe a workload archive acquired at the science-gateway level, and we show its added value on several case studies related to user accounting, pilot jobs, fine-grained task analysis, bag of tasks, and workflows. Results show that science-gateway workload archives can detect workload wrapped in pilot jobs, improve user identification, give information on distributions of data transfer times, make bag-of-task detection accurate, and retrieve characteristics of workflow executions. Some limits are also identified.

1 Introduction

Grid workload archives [1–5] are widely used for research on distributed systems, to validate assumptions, to model computational activity [6, 7], and to evaluate methods in simulation or in experimental conditions. Available workload archives are acquired at the infrastructure level, by computing sites or by central monitoring and bookkeeping services. However, user communities often access the infrastructure through stacks of application-level middleware such as workflow engines, application wrappers, pilot-job systems, and portals. As a result, workload archives lack critical information about dependencies among tasks, about task sub-steps, about artifacts introduced by application-level scheduling, and about users. Methods have been proposed to recover this information. For instance, [8] detects bags of tasks as tasks submitted by a single user in a given time interval. In other cases, information can hardly be recovered: [2] reports that there is currently no study of a pilot-job workload, and workflow studies such as [5] are mostly limited to test runs conducted by developers.

Meanwhile, science gateways are emerging as user-level platforms to access distributed infrastructures. They combine a set of authentication, data transfer, and workload management tools to deliver computing power as transparently as possible. They are used by groups of users over time, and therefore gather rich information about workload patterns. The Virtual Imaging Platform (VIP) [9],

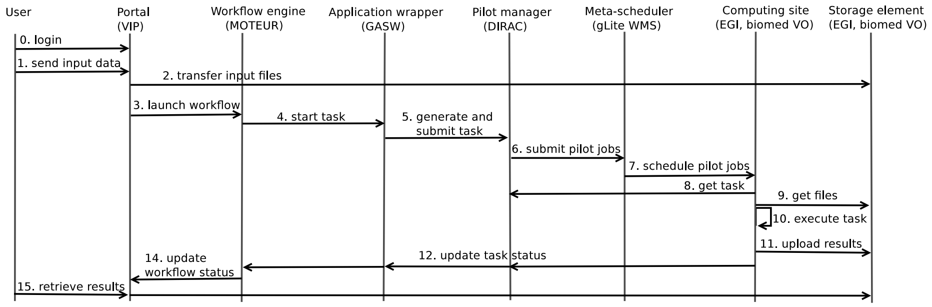

Fig. 1. Considered science-gateway architecture. Tools in brackets are used here.

for instance, is an open web platform for medical simulation. Other examples include e-bioinfra [10], the P-Grade portal [11], the Science-Gateway framework in [12], MediGRID [13], and CBRAIN[1].

This paper describes a science-gateway workload archive, and illustrates its added value to archives acquired at the infrastructure level. The model is presented in Section 2 and used in 5 case studies in Section 3: Section 3.1 studies pilot jobs, Section 3.2 compares user accounting to data acquired by the infrastructure, Section 3.3 performs fine-grained task analysis, Section 3.4 evaluates the accuracy of bag of task detection from infrastructure-level traces, and Section 3.5 analyzes workflows in production. Section 4 concludes the paper.

2 A Science-Gateway Workload Archive

Science gateways usually involve a subset or all the entities shown on Fig. 1, which describes the VIP architecture used here. Users authenticate to a web portal with login and password, and they are then mapped to X.509 robot credentials. From the portal, users mainly transfer data and launch workflows executed by an engine. The engine has an application wrapper which generates tasks from application descriptions and submits them to a pilot-job scheduler. In a pilot-job model [14–16], generic pilot jobs are submitted to the infrastructure instead of application tasks. When these jobs reach a computing site, they fetch tasks from the pilot manager. Tasks then download input files, execute, and upload their results. To increase reliability and performance, tasks can also be replicated as described in [17]. Task replicas may also be aborted to limit resource waste. This science gateway model totally applies to e-bioinfra, and partly to the P-Grade portal, the Science-Gateway framework in [12], medigrid-DE, and CBRAIN.

Our science-gateway archive model adopts the schema on Fig. 2. `Task` contains information such as final status, exit code, timestamps of internal steps, application and workflow activity name. Each task is associated to a `Pilot Job`. `Workflow Execution` gathers all the activities and tasks of a workflow execution, `Site` connects pilots and tasks to a grid site, and `File` provides the list of

[1] http://cbrain.mcgill.ca

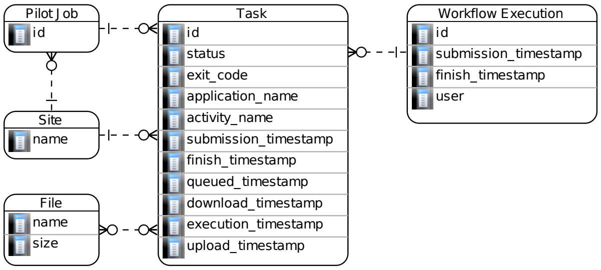

Fig. 2. Science-gateway archive model.

files associated to a task and workflow execution. In this work we focus on `Task`, `Workflow Execution` and `Pilot Job`.

The science-gateway archive is extracted from VIP. `Task`, `Site` and `Workflow Execution` information are acquired from databases populated by the workflow engine at runtime. `File` and `Pilot Job` information are extracted from the parsing of task standard output and error files.

Studies presented in the following Sections are based on the workload of the VIP from January 2011 to April 2012. It consists of 2,941 workflow executions, 112 users, 339,545 pilot jobs, 680,988 tasks where 338,989 are completed tasks, 138,480 error tasks, 105,488 aborted tasks, 15,576 aborted task replicas, 48,293 stalled tasks and 34,162 submitted or queued tasks. Stalled tasks are tasks which lost communication with the pilot manager, e.g. because they were killed by computing sites due to quota violation. Tasks ran on the biomed virtual organization of the European Grid Infrastructure (EGI[2]). Traces used in this work are available to the community in the Grid Observatory[3].

3 Case studies

3.1 Pilot jobs

Pilot jobs are increasingly used to improve scheduling and reliability on production grids [14–16]. This type of workload, however, is difficult to analyze from infrastructure traces as a single pilot can wrap several tasks, which remains unknown to the infrastructure. In our case, pilots are discarded after 5 task executions, if the remaining walltime allowed on the site cannot be obtained, if they are idle for more than 10 minutes, or if one of their tasks fails. Pilots can execute any task submitted by the science gateway, regardless of the workflow execution and user.

Fig. 3 shows the number of tasks and users per pilot in the archive. Out of the 646,826 executed tasks in the archive, only those for which a standard output containing the pilot id could be retrieved are represented. This corresponds to

[2] http://www.egi.eu
[3] http://www.grid-observatory.org

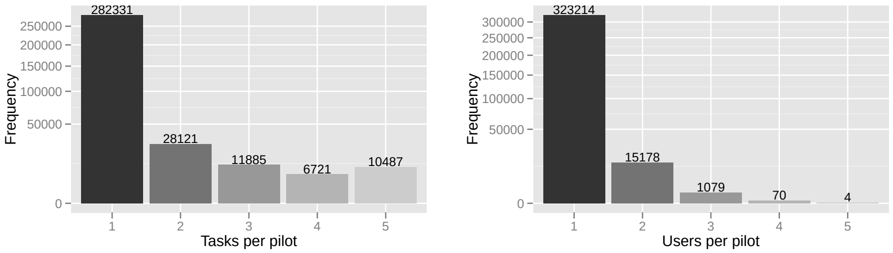

Fig. 3. Histogram of tasks per pilot (left) and users per pilot (right).

453,547 tasks, i.e. 70% of the complete task set. Most pilots (83%) execute only 1 task due to walltime limits or other discards. These 83% execute 282,331 tasks, which represents 62% of the considered tasks. Workload acquired at the infrastructure level would usually assimilate pilot jobs to tasks. Our data shows that this hypothesis is only true for 62% of the tasks. The distribution of users per pilots has a similar decrease: 95% of the pilots execute tasks of a single user.

3.2 Accounting

On a production platform like EGI, accounting data consists of the list of active users and their number of submitted jobs, consumed CPU time, and wall-clock time. Here, we compare data provided by the infrastructure-level accounting services of EGI[4] to data obtained from the science-gateway archive.

Fig. 4 compares the number of users reported by EGI and the scientific gateway. It shows a dramatic discrepancy between the two sources of information, explained by the use of a robot certificate in the gateway. Robot certificates are regular X.509 user certificates that are used for all grid operations performed by a science gateway, namely data transfers and task submission. From an EGI point of view, all VIP users are accounted as a single user regardless of their real identity. EGI reports more than one user for months 12, 13, 15 and 16 due to updates of the VIP certificate. The adoption of robot certificates totally discards the accounting of user names at the infrastructure level. Studies such as presented on Fig. 17 in [18] or on Fig. 1 in [2], cannot be considered reliable in this context. Robot certificates are not an exception: a survey available online[5] shows that 80 of such certificates are known on EGI. By avoiding the need for users to request personal certificates, they simplify the access to the infrastructure to a point that their very large adoption in science gateways seems unavoidable.

Fig. 5 compares the number of submitted jobs, consumed CPU time, and consumed wall-clock time obtained by the EGI infrastructure and by VIP. The number of jobs reported by EGI is almost twice as important as in VIP. This huge discrepancy is explained by the fact that many pilot jobs do not register to the pilot system due to some technical issues, or do not execute any task due to the

[4] http://accounting.egi.eu

[5] https://wiki.egi.eu/wiki/EGI_robot_certificate_users

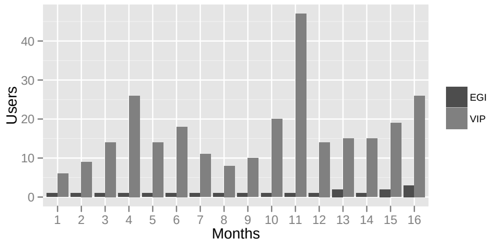

Fig. 4. Number of reported EGI and VIP users.

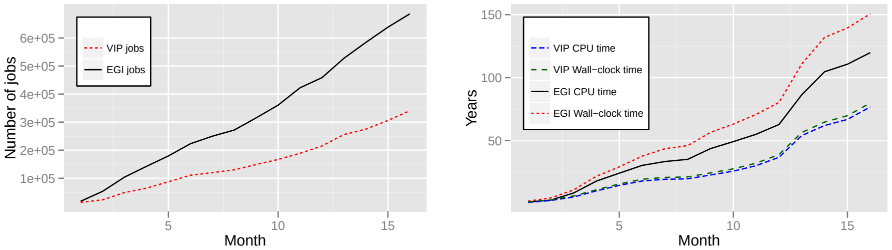

Fig. 5. Number of submitted pilot jobs (*left*), and consumed CPU and wall-clock time (*right*) by the infrastructure (EGI) and the science gateway (VIP).

absence of workload, or execute tasks for which no standard output containing the pilot id could be retrieved. These pilots cannot be identified from the task logs. While this highlights serious potential improvements in the pilot manager, it also reveals that a significant fraction of the workload measured by EGI does not come from applications but from artifacts introduced by pilot managers. This should be taken into account when conducting studies on application-level schedulers from workload acquired at the infrastructure level.

About 60 walltime years are missing from the science gateway archive, compared to the infrastructure. This is due to the pilot setup time (a few minutes per pilot), and to the computing time of lost tasks, for which no standard output containing monitoring data could be retrieved. Tasks are lost (a.k.a stalled) in case of technical issues such as network interruption or deliberate kill from sites due to quota violation. Better investigating this missing information is part of our future work.

3.3 Task analysis

Traces acquired at the science-gateway level provide fine-grained information about tasks, which is usually not possible at the infrastructure level. Fig. 6 shows the distributions of download, upload and execution times for successfully completed tasks. Distributions show a substantial amount of very long steps.

Error causes can also be investigated from science-gateway archives. Fig. 7 (left) shows the occurrence of 6 task-level errors. These error codes are application-specific and not accessible to infrastructure level archives, see e.g. [19] (Table 3) and [3].

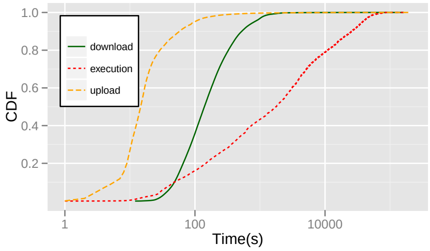

Fig. 6. Different steps in task life.

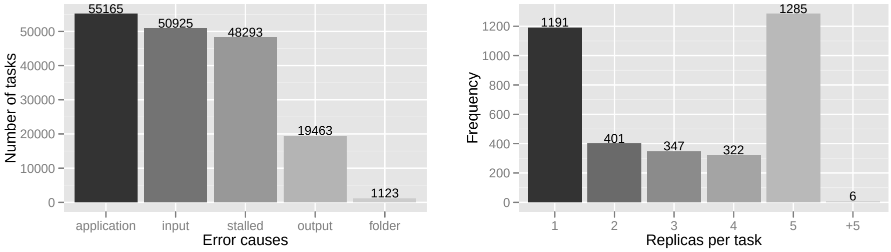

Fig. 7. Task error causes (*left*) and number of replicas per task (*right*).

A common strategy to cope with recoverable errors is to replicate tasks [20], which is usually not known to the infrastructure. Fig. 7 (right) shows the occurrence of task replication in the science-gateway archive.

3.4 Bag of tasks

In this section, we evaluate the accuracy of the method presented in [8] to detect bag of tasks (BoT). This method considers that two tasks successively submitted by a user belong to the same BoT if the time interval between their submission times is lower or equal to a time Δ. The value of Δ is set to $120s$ as described in [8]. Fig. 8 presents the impact of Δ on BoT sizes (a.k.a. batch sizes) for $\Delta = 10s, 30s, 60s$ and $120s$.

Fig. 9 presents the comparison of BoT characteristics obtained from the described method for $\Delta = 120s$ and from VIP. BoTs in VIP were extracted as the tasks generated by the same activity in a workflow execution. Thus, they can be considered as ground truth and are named `Real Non-Batch` for single-task BoTs and `Real Batch` for others. Analogously, we name `Non-Batch` and `Batch` BoTs determined by the method. `Batch` has about 90% of its BoT sizes ranging from 2 to 10 while these batches represent about 50% of `Real Batch`. This discrepancy has a direct impact on the BoT duration (makespan), inter-arrival time and consumed CPU time. The duration of `Non-Batch` are overestimated up to 400%, inter-arrival times for both `Batch` and `Non-Batch` are underestimated by about 30% in almost all intervals, and consumed CPU times are underestimated of 25% for `Non-Batch` and of about 20% for `Batch`.

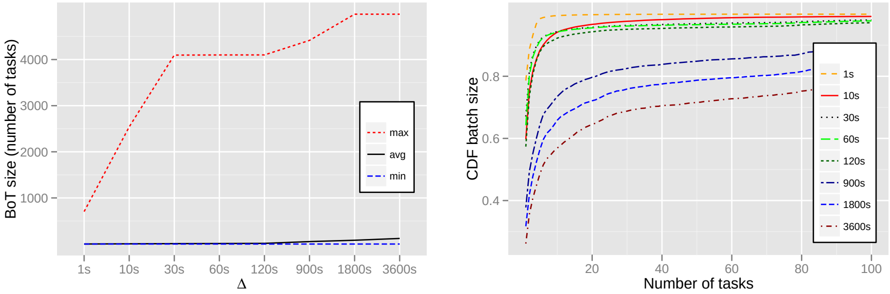

Fig. 8. Impact of parameter Δ on BoT sizes: minimum, maximum and average values (*left*); and its distribution (*right*).

This data shows that detecting bag of tasks based on infrastructure-level traces is very inaccurate. Such inaccuracy may have important consequences on works based on such detection, e.g. [21].

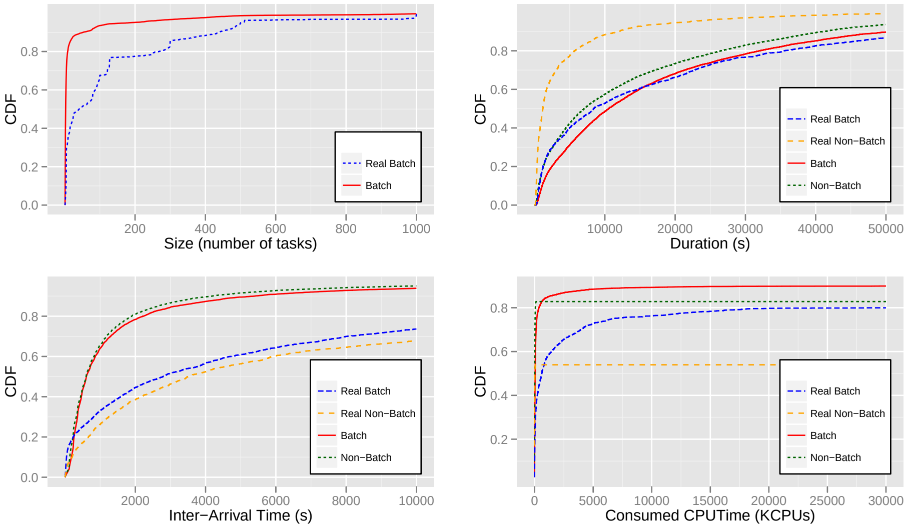

Fig. 9. CDFs of characteristics of batched and non-batched submissions: *BoT sizes*, *duration per BoT*, *inter-arrival time* and *consumed CPU time* ($\Delta = 120s$).

3.5 Workflows

Few works study the characterization of grid workflow executions. In [5], the authors present the characterization of 2 workloads that are mostly test runs conducted by developers. To the best of our knowledge, there is no work on the characterization of grid workflows in production.

Fig. 10 presents characteristics of the workflow executions extracted from our science-gateway archive. They could be used to build workload generators for the evaluation of scheduling algorithms. Let N be the number of tasks in a workflow execution; we redefine the 3 classes presented in [5] to `small` for $N \leq 100$, `medium` for $100 < N \leq 500$ and `large` for $N > 500$. From Fig. 10 (top left), we observe that the workload is composed by $52\%, 31\%$ and 17% of `small`, `medium` and `large` executions respectively. In Fig.10 (top right), 90% of `small`, 66% of `medium` and 54% of `large` executions have a makespan lower than 14 hours. Speedup values presented in Fig. 10 (bottom left) show that execution speed-up increases with the size of the workflow, which indicates good parallelization. Critical path lengths are mostly up to 2 levels for `small` and `large` executions and up to 3 for `medium` executions (bottom right of Fig. 10).

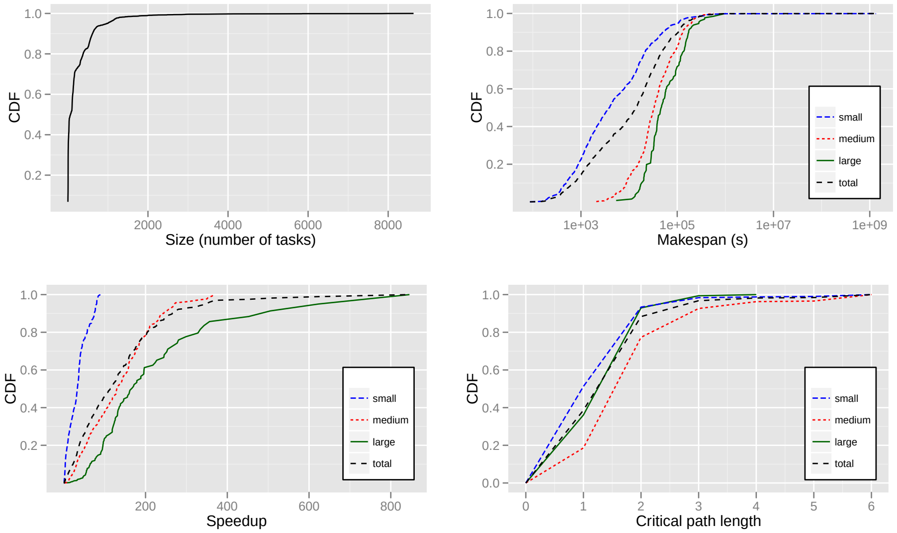

Fig. 10. Characteristics of workflow executions: number of tasks (*top left*), CPU time and makespan (*top right*), speedup (*bottom left*) and critical path length (*bottom right*).

4 Conclusion

We presented a science-gateway model of workload archive containing detailed information about users, pilot jobs, task sub-steps, bag of tasks and workflow executions. We illustrated the added value of science-gateway workloads compared to infrastructure-level traces using information collected by the Virtual Imaging Platform in 2011/2012, which consist of $2,941$ workflow executions, 339,545 pilot jobs, 680,988 tasks and 112 users that consumed about 76 CPU years.

Several conclusions demonstrate the added-value of a science-gateway approach to workload archives. First, it can exactly identify tasks and users, while infrastructure-level traces cannot identify 38% of the tasks due to their bundling in pilot jobs, and cannot properly identify users when robot certificates are used. Infrastructure archives are also hampered by additional workload artifacts coming from pilot-job schedulers, which can be distinguished from application workload using science-gateway archives. More detailed information about tasks is also available from science-gateway traces, such as distributions of download, upload and execution times, and information about replication. Besides, the detection of bag of tasks from infrastructure traces is inaccurate, while a science-gateway contains ground truth. Finally, we reported a few parameters on workflow executions, which could not be extracted from infrastructure-level traces. Limits of science-gateway workloads still exist. In particular, it is very common that a significant fraction of lost tasks do not report complete information.

Traces acquired by the Virtual Imaging Platform will be regularly made available to the community in the Grid Observatory. We hope that other science-gateway providers could also start publishing their traces so that computer-science studies can better investigate production conditions. Information provided by such science-gateway archives can be used, to elaborate benchmarks, to simulate applications and algorithms targeting production systems, or to feed algorithms with historical information [17].

Studies presented in this work only show a partial overview of the potential of science-gateway traces. In particular, information about file access pattern, about the number and location of computing sites used per workflow or bag-of-task execution, and about task resubmission is available in the archive.

5 Acknowledgment

This work is funded by the French National Agency for Research under grant ANR-09-COSI-03 "VIP". We thank the European Grid Initiative and National Grid Initiatives, in particular France-Grilles, for the infrastructure and support.

References

1. Iosup, A., Li, H., Jan, M., Anoep, S., Dumitrescu, C., Wolters, L., Epema, D.H.J.: The grid workloads archive. Future Gener. Comput. Syst. **24**(7) (2008) 672–686
2. Iosup, A., Epema, D.: Grid computing workloads: bags of tasks, workflows, pilots, and others. Internet Computing, IEEE **15**(2) (march-april 2011) 19 –26
3. Kondo, D., Javadi, B., Iosup, A., Epema, D.: The failure trace archive: Enabling comparative analysis of failures in diverse distributed systems. In: CCGrid 2010. (2010) 398 –407
4. Germain-Renaud, C., Cady, A., Gauron, P., Jouvin, M., Loomis, C., Martyniak, J., Nauroy, J., Philippon, G., Sebag, M.: The grid observatory. IEEE International Symposium on Cluster Computing and the Grid (2011) 114–123
5. Ostermann, S., Prodan, R., Fahringer, T., Iosup, R., Epema, D.: On the characteristics of grid workflows. In: CoreGRID Symposium - Euro-Par 2008. (2008)
6. Christodoulopoulos, K., Gkamas, V., Varvarigos, E.: Statistical analysis and modeling of jobs in a grid environment. Journal of Grid Computing **6** (2008) 77–101

7. Medernach, E.: Workload analysis of a cluster in a grid environment. In: Job Scheduling Strategies for Parallel Processing. (2005) 36–61
8. Iosup, A., Jan, M., Sonmez, O., Epema, D.: The characteristics and performance of groups of jobs in grids. In: Euro-Par. (2007) 382–393
9. Ferreira da Silva, R., Camarasu-Pop, S., Grenier, B., Hamar, V., Manset, D., Montagnat, J., Revillard, J., Balderrama, J.R., Tsaregorodtsev, A., Glatard, T.: Multi-Infrastructure Workflow Execution for Medical Simulation in the Virtual Imaging Platform. In: HealthGrid 2011, Bristol, UK (2011)
10. Shahand, S., Santcroos, M., Mohammed, Y., Korkhov, V., Luyf, A.C., van Kampen, A., Olabarriaga, S.D.: Front-ends to Biomedical Data Analysis on Grids. In: Proceedings of HealthGrid 2011, Bristol, UK (june 2011)
11. Kacsuk, P.: P-GRADE Portal Family for Grid Infrastructures. Concurrency and Computation: Practice and Experience **23**(3) (2011) 235–245
12. Ardizzone, V., Barbera, R., Calanducci, A., Fargetta, M., Ingrà, E., La Rocca, G., Monforte, S., Pistagna, F., Rotondo, R., Scardaci, D.: A european framework to build science gateways: architecture and use cases. In: 2011 TeraGrid Conference: Extreme Digital Discovery, New York, ACM (2011) 43:1–43:2
13. Krefting, D., Bart, J., Beronov, K., Dzhimova, O., Falkner, J., Hartung, M., Hoheisel, A., Knoch, T.A., Lingner, T., Mohammed, Y., Peter, K., Rahm, E., Sax, U., Sommerfeld, D., Steinke, T., Tolxdorff, T., Vossberg, M., Viezens, F., Weisbecker, A.: Medigrid: Towards a user friendly secured grid infrastructure. Future Generation Computer Systems **25**(3) (2009) 326 – 336
14. Luckow, A., Weidner, O., Merzky, A., Maddineni, S., Santcroos, M., Jha, S.: Towards a common model for pilot-jobs. In: HPDC12, Delft, The Netherlands (2012)
15. Tsaregorodtsev, A., Brook, N., Ramo, A.C., Charpentier, P., Closier, J., Cowan, G., Diaz, R.G., Lanciotti, E., Mathe, Z., Nandakumar, R., Paterson, S., Romanovsky, V., Santinelli, R., Sapunov, M., Smith, A.C., Miguelez, M.S., Zhelezov, A.: DIRAC3 . The New Generation of the LHCb Grid Software. Journal of Physics: Conference Series **219**(6) (2009) 062029
16. Thain, D., Tannenbaum, T., Livny, M.: Distributed computing in practice: the condor experience. Concurrency and Computation: Practice and Experience **17**(2-4) (2005) 323–356
17. Ferreira da Silva, R., Glatard, T., Desprez, F.: Self-healing of operational workflow incidents on distributed computing infrastructures. In: IEEE/ACM CCGrid 2012, Ottawa, Canada (2012) 318–325
18. Ilijasic, L., Saitta, L.: Characterization of a Computational Grid as a Complex System. In: Grid Meets Autonomic Computing(GMAC'09). (June 2009) 9–18
19. Lingrand, D., Montagnat, J., Martyniak, J., Colling, D.: Optimization of jobs submission on the EGEE production grid: modeling faults using workload. Journal of Grid Computing (JOGC) Special issue on EGEE **8**(2) (March 2010) 305–321
20. Casanova, H.: On the harmfulness of redundant batch requests. International Symposium on High-Performance Distributed Computing **0** (2006) 255–266
21. Brasileiro, F., Gaudencio, M., Silva, R., Duarte, A., Carvalho, D., Scardaci, D., Ciuffo, L., Mayo, R., Hoeger, H., Stanton, M., Ramos, R., Barbera, R., Marechal, B., Gavillet, P.: Using a simple prioritisation mechanism to effectively interoperate service and opportunistic grids in the eela-2 e-infrastructure. Journal of Grid Computing **9** (2011) 241–257