Journal of Physics: Conference Series **513** (2014) 032093 doi:10.1088/1742-6596/513/3/032093

Self managing experiment resources

F Stagni[1], M Ubeda[1], A Tsaregorodtsev[3], V Romanovskiy[4], S Roiser[5], P Charpentier[1], R Graciani[2]

[1]PH Department, CH-1211 Geneva 23 Switzerland
[2]Universitat de Barcelona, Gran Via de les Corts Catalanes, 585 Barcelona
[3]C.P.P.M - 163, avenue de Luminy - Case 902 - 13288 Marseille cedex 09
[4]IHEP, Protvino
[5]IT Department, CH-1211 Geneva 23 Switzerland

E-mail: federico.stagni@cern.ch

Abstract. Within this paper we present an autonomic Computing resources management system, used by LHCb for assessing the status of their Grid resources. Virtual Organizations Grids include heterogeneous resources. For example, LHC experiments very often use resources not provided by WLCG, and Cloud Computing resources will soon provide a non-negligible fraction of their computing power. The lack of standards and procedures across experiments and sites generated the appearance of multiple information systems, monitoring tools, ticket portals, etc... which nowadays coexist and represent a very precious source of information for running HEP experiments Computing systems as well as sites. These two facts lead to many particular solutions for a general problem: managing the experiment resources. In this paper we present how LHCb, via the DIRAC interware, addressed such issues. With a renewed Central Information Schema hosting all resources metadata and a Status System (Resource Status System) delivering real time information, the system controls the resources topology, independently of the resource types. The Resource Status System applies data mining techniques against all possible information sources available and assesses the status changes, that are then propagated to the topology description. Obviously, giving full control to such an automated system is not risk-free. Therefore, in order to minimise the probability of misbehavior, a battery of tests has been developed in order to certify the correctness of its assessments. We will demonstrate the performance and efficiency of such a system in terms of cost reduction and reliability.

1. Introduction

DIRAC[1][2] is a community Grid solution. Developed in python, it offers powerful job submission functionalities, and a developer-friendly environment to create services and agents. Services expose an extended Remote Process Call (RPC) and Data Transfer implementation; agents are a stateless light-weight component, comparable to a cron-job. Being a community (also addressed as Virtual Organization, or VO) Grid solution, DIRAC can interface with many resource types and providers. Resource types are, for example, a *computing element* (CE), or a *catalog*. Within the same resource types, different implementations are developed for different providers. DIRAC acts as an abstraction layer for interfacing many different resources in a similar way. It also gives the opportunity to instantiate DIRAC types of CEs, *storage elements* (SEs), or catalogs. It is possible to access resources at certain type of clouds [3], and new types

of resources can be easily plugged in, like recently happened, in LHCb, with "vacuum" resources, and BOINC volunteer computing.

Every VO needs to administer the resources it is using. This means knowing their status, and reacting accordingly. For example, it makes no sense to try and use a CE that is aborting all the jobs, or trying to write to a SE where all the free space is exhausted. For these cases, the administrator is better banning them. At the same time, using effectively the resources means putting them back in production once they recover. In practice, a good administrator should follow closely the available monitoring information.

Using DIRAC, a community can effectively make use of different types of resources, from different providers, in a seamless way. Each resource needs to be administered, and for this scope monitoring tools are available. Often, different resource providers expose also different monitoring information. The more the VO resources grow, both in number and in type, the more difficult it becomes, for the administrator, to manage them. This is why, within DIRAC, there are common views for all the resources. Common view means that the administration of each resource is done using only one tool.

This paper explains, in section 2, how DIRAC abstracts the description of different types of resources, developed by different providers. Section 3 explains how, within DIRAC, we provide services and agents that can assess the status of the resources in an automated way, easing the work of a community administrator. Section 4 shows the adoption of such automated tools within LHCb, and some results from the past year. The final remarks, and some notes about future works, are given in section 5.

2. Resources ontology

Figure 1 shows graphically the generic ontology adopted by DIRAC to abstract the description of the resources. The main idea is to structure it following the logic of the resource provisioning. Therefore, it is based on the notion of a *Site* as the main body responsible for the services offered to the user communities. Organizing resources by Sites gives a clear administrative information about whom to contact when needed. At the same time it provides a natural proximity relation between different types of resources that is essential for DIRAC in order to, for example, optimize the jobs scheduling.

The adopted schema is based on the following key concepts:

- **Community** (or VO): is a group of users that use the resources in a coordinated way. Each Community may define its own policies for accessing and using the resources, within the limits allowed by the Sites.
- **Site**: is the central piece of the new schema both from a functional and an administrative point of view. It collects access points to resources that are related by locality in a functional sense, i.e. the storage at a given Site is considered local to the CPU at the same Site and this relation will be used by DIRAC. In DIRAC, a Site can be from a fraction of physical computer center, to a whole regional Grid. It is the responsibility of the DIRAC administrator to properly define the Sites. Not all Sites need to grant access to all VOs supported in the DIRAC installation.
- **Domain**: is a supra-site organization meaningful in the context of a given DIRAC installation. Domains might be related to the funding of the resources (GISELA, WLCG,...), to administrative domains (NGIs) or any other reason relevant for the installation. They have no functional meaning in DIRAC and are only used for reporting purposes.
- **Resource type** is the main category in which relevant IT Resources are grouped. At the moment the following types are relevant:
  - **Computing**: contains information about the interfaces used to access Computing resources at the Site. A subsection of a computing element contains, for example,

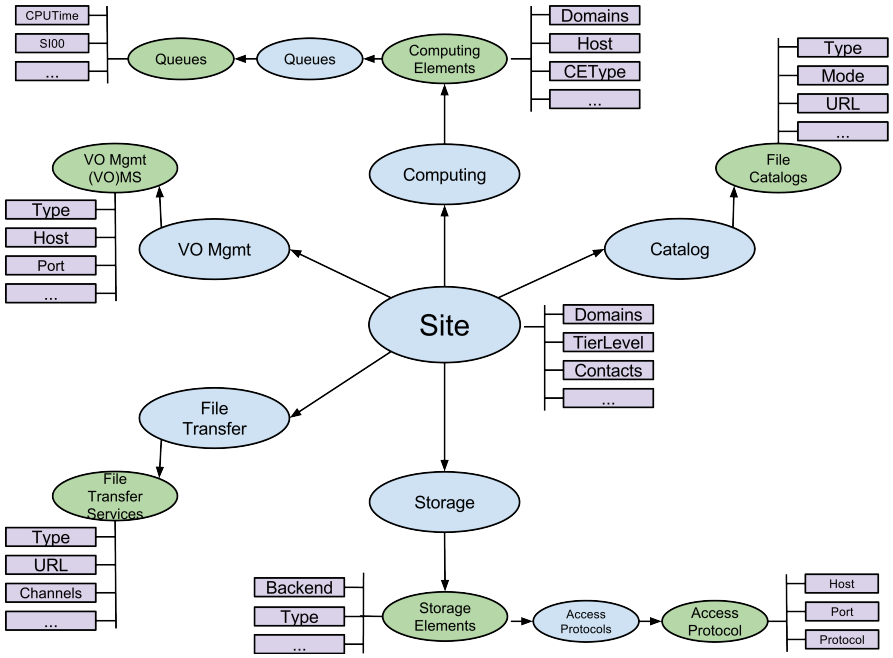

Figure 1. In DIRAC, all the Grid Resources, independently of their types, are abstracted in a single, generic ontology

information about the exposed queues.
- **Storage**: contains information about the interfaces to access Storage resources at the Site. The access protocols are specified as a subsection.
- **Catalog**: contains description of the available File Catalogs. This includes third party catalogs, but also DIRAC File Catalogs [4].
- **File Transfer**: includes the description of third party transfer services.
- **Databases**: describes instances of the database servers available in the installation.
- **VO Management**: describes Management services like the VOMS servers.

Such ontology is persisted in the DIRAC Configuration Service (CS), and as such it is constantly available to all DIRAC components.

3. The DIRAC Resource Status System (RSS)

The DIRAC Resource Status System (RSS, from now on) is an autonomous policy system acting as a central status information point for DIRAC resources, that enforces managerial and operational actions automatically. The status of an entity can be evaluated using a number of policies, each making assessments relative to specific monitoring information. Individual results of these policies can be combined to evaluate and propose a global status for the entity. External monitoring and testing systems are used by policies for site commission and certification.

The RSS caches dispersed monitoring information, depending on the type and nature of the resources. Examples of cached information are the official DownTime of WLCG resources, as they are announced in the Grid Operation Center (GOC) DB. The RSS can also cache the efficiency of jobs and pilot jobs (a well know pull-type of jobs), as well as the storage element occupancies. From an operational perspective, the community administrators define which policies to apply, depending on the type of resources. With the collected information, and the given current status, the policies are evaluated, and a status for each resource is reassessed. The definition of policies is easily done within the DIRAC Configuration Service.

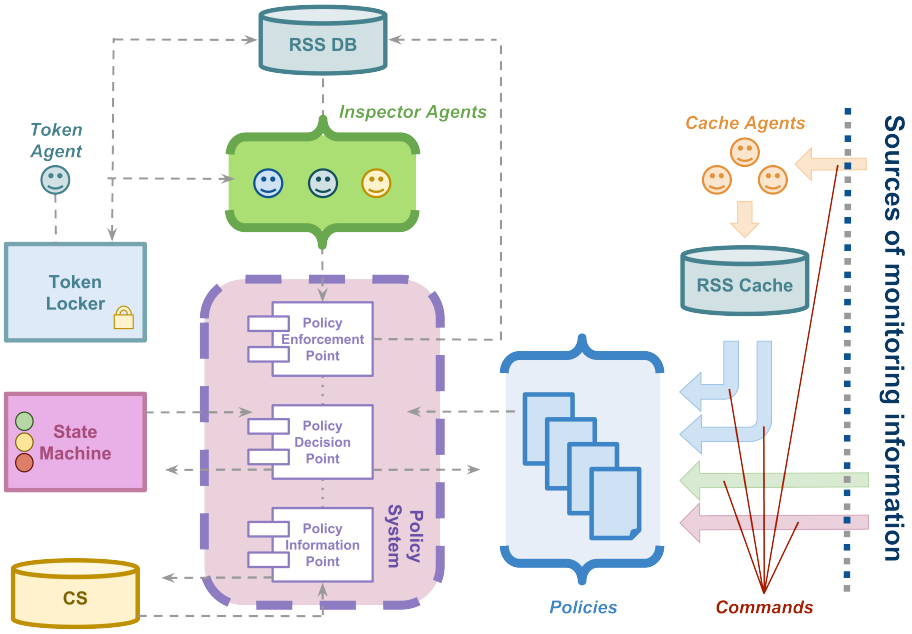

Figure 2. A policy system machinery is used to assess the resources status

Within figure 2, we can see, schematically, the policy system machinery of the RSS. Interested readers can find more details in [5].

4. Adoption and results within LHCb

The policies listed below are all part of the policies set adopted within LHCb:

- A WLCG site announces a DownTime (DT) in the GOC DB: the site is banned n hours in advance (n is fully configurable).
- That same site exits from the DT: tests jobs can be sent to the site. In case these tests are positive, the site is unbanned.
- A Storage Element (SE) runs out of space: the SE is banned for writing.
- There are certain SEs from which we do not allow to remove any data.

This simple set of policies is just an example of what can be done with a generic policy system.

Within figure 3, we show a comparison between the reaction time of a shifter, prior to the adoption of the policies, and the RSS. The enhancement, as expected, is very well visible.

5. Conclusions and future work

The development of the RSS, as well as the definition of the resources ontology, took a substantial amount of time. The experience matured within the years helped in creating a system that has proven to be scalable, and reliable. LHCb has developed the RSS, and it has used it in production since many months, in the framework of the DIRAC Interware. Being DIRAC a fully VO-agnostic framework, its adoption by other VOs is easy. VO-specifics are instead the policies themselves, as have been written in the LHCb extension of DIRAC [6].

In its essence, the RSS is stable, scalable, and reliable: for these reasons, we do not foresee any substantial changes. LHCb plans to introduce new policies in its production installation, so that it will become more and more easier for our community to take full advantage of our resources.

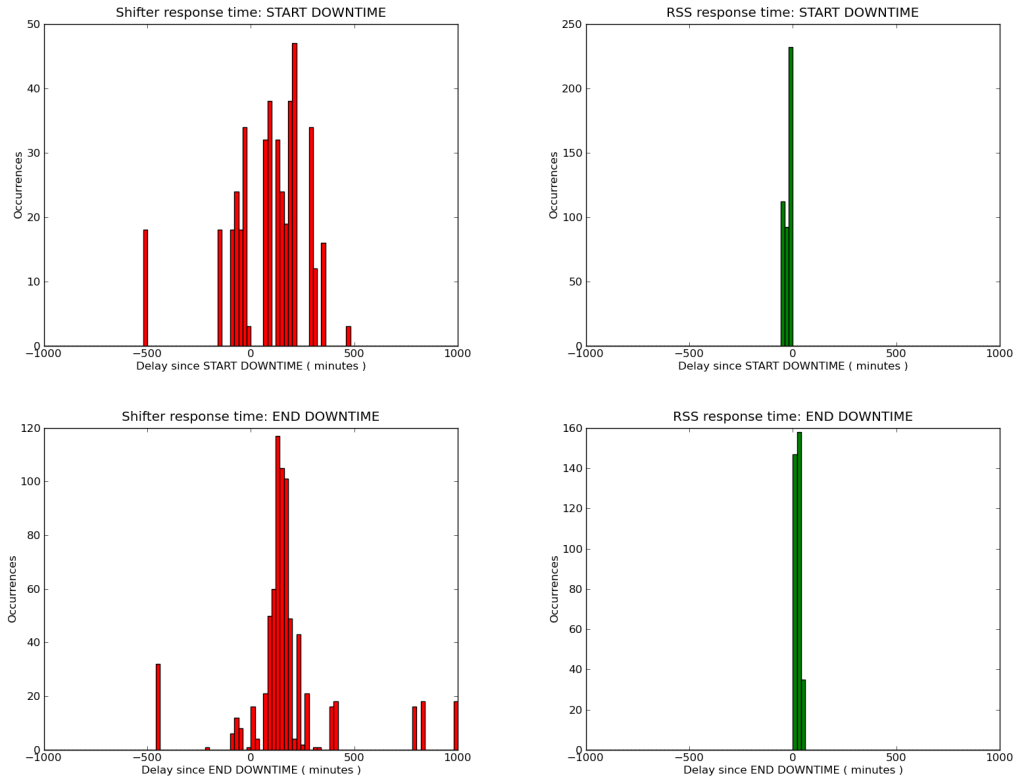

Figure 3. Comparison of the reaction times from the official start and end of a DownTime, and ban/unban of Storage Elements. The 3 different peaks registered for RSS are due to changes in the policy conditions over its application period.

References

[1] Casajus A, Ciba K, Fernandez V, Graciani R, Hamar V, Mendez V, Poss S, Sapunov M, Stagni F, Tsaregorodtsev A and Ubeda M 2012 *Journal of Physics: Conference Series* **396** 032107 URL `http://stacks.iop.org/1742-6596/396/i=3/a=032107`

[2] 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 and Zhelezov A 2010 *Journal of Physics: Conference Series* **219** 062029

[3] Muoz V M, Albor V F, Diaz R G, Ramo A C, Pena T F, Arvalo G M and Silva J J S 2012 *Journal of Physics: Conference Series* **396** 032075 URL `http://stacks.iop.org/1742-6596/396/i=3/a=032075`

[4] Tsaregorodtsev A and Poss S 2012 *Journal of Physics: Conference Series* **396** 032108 URL `http://stacks.iop.org/1742-6596/396/i=3/a=032108`

[5] Ubeda M, Stagni F, Tsaregorodtsev A, Charpentier P and Bernardoff V 2012 *Journal of Physics: Conference Series* **396** 032110 URL `http://stacks.iop.org/1742-6596/396/i=3/a=032110`

[6] Stagni F, Charpentier P, Graciani R, Tsaregorodtsev A, Closier J, Mathe Z, Ubeda M, Zhelezov A, Lanciotti E, Romanovskiy V, Ciba K D, Casajus A, Roiser S, Sapunov M, Remenska D, Bernardoff V, Santana R and Nandakumar R 2012 *Journal of Physics: Conference Series* **396** 032104 URL `http://stacks.iop.org/1742-6596/396/i=3/a=032104`