An Innovative Science Gateway for the Cherenkov Telescope Array

Alessandro Costa · Pietro Massimino · Marilena Bandieramonte · Ugo Becciani · Mel Krokos · Costantino Pistagna · Simone Riggi · Eva Sciacca · Fabio Vitello

Abstract The Cherenkov Telescope Array (CTA) is currently building the next generation, ground-based, very high-energy gamma-ray instrumentation. CTA is expected to collect very large datasets (in the order of petabytes) which will have to be stored, managed and processed. This paper presents a graphical user interface built inside a science gateway aiming at providing CTA-users with a common working framework.

Alessandro Costa
INAF Astrophysical Observatory of Catania, via S. Sofia 78
95123 Catania, Italy Tel.: +39 095 7332273
E-mail: alessandro.costa@oact.inaf.it

Pietro Massimino
INAF Astrophysical Observatory of Catania,
E-mail: piero.massimino@oact.inaf.it

Marilena Bandieramonte
Dipartimento di Fisica e Astronomia, University of Catania, Italy
E-mail: marilena.bandieramonte@oact.inaf.it

Ugo Becciani
INAF Astrophysical Observatory of Catania,
E-mail: ugo.becciani@oact.inaf.it

Mel Krokos
University of Portsmouth, UK,
E-mail: Mel.Krokos@port.ac.uk

Costantino Pistagna
INAF Astrophysical Observatory of Catania,
E-mail: costantino.pistagna@oact.inaf.it

Simone Riggi
INAF Astrophysical Observatory of Catania,
E-mail: simone.riggi@oact.inaf.it

Eva Sciacca
INAF Astrophysical Observatory of Catania,
E-mail: eva.sciacca@oact.inaf.it

Fabio Vitello
INAF Astrophysical Observatory of Catania,
E-mail: fabio.vitello@oact.inaf.it

The gateway is WS-PGRADE/gUSE workflow-oriented and is equipped with a flexible SSO (based on SAML) to control user access for authentication and authorization. An interactive desktop environment is provided, called Astronomical & Physics Cloud Interactive Desktop (ACID). Users are able to exploit the graphical interface as provided natively by the tools included in ACID. A cloud data service shares and synchronizes data files and output results between the user desktop and the science gateway. Our solution is a first attempt towards an ecosystem of new technologies with a high level of flexibility to suit present and future requirements of the CTA community.

Keywords CTA · Science Gateways · Workflows · gUSE · WS-PGRADE · Cloud Services · ACID

1 Introduction

The present generation of imaging atmospheric Cherenkov telescopes, e.g. H.E.S.S.[1], MAGIC[2] and VERITAS[3], has in recent years opened the realm of ground-based gamma ray astronomy in the energy range above a few tens of GeV. The Cherenkov Telescope Array (CTA)[4] will explore our Universe in depth in very high energy (> 10 GeV) gamma-rays and will investigate cosmic non-thermal processes, in close cooperation with observatories operating at other wavelength ranges of the electromagnetic spectrum, and those using other messenger particles such as cosmic rays and neutrinos.

This scientific gateway aims at providing an appropriate Graphical User Interface (GUI) to access e-

---

[1] `http://www.mpi-hd.mpg.de/hfm/HESS/`
[2] `https://magic.mpp.mpg.de/`
[3] `http://veritas.sao.arizona.edu/`
[4] `https://portal.cta-observatory.org/`

science and data management monitoring tools within a common working ecosystem. Recent developments in scientific gateways [1],[2] allow users to store, manage and share large data collections. This technology offers scientists an environment so that they can focus on the actual applications instead of learning and managing the required infrastructures to execute them.

This paper presents a scientific gateway for the CTA community based on the workflow-oriented framework WS-PGRADE/gUSE [3]. This framework supports operation and maintenance of a generic-purpose science gateway that cat be highly customized and can access HTC infrastructures, data and networking infrastructures and services e.g. grids, desktop grids, clusters, supercomputers, [4] but also academic and commercial clouds.

We designed and implemented an application specific gateway tailored to the CTA user community. Our first release, presented in this paper, opens new possibilities for federated authentication and authorization functionality. We implemented a web browser Single Sign On (SSO) profile using Shibboleth 2.0 [5] which is an open-source tool for identity management and federated authentication and authorization based on SAML 2.0 [6]. We provide a new set of services in the form of an interactive desktop called Astronomical Physics Cloud Interactive Desktop (ACID). This approach gives users the possibility to exploit natively the graphical user interface of any included tools. A large number of tools is currently available, e.g. the Heasoft Unified Release with over 100 tools (such as Ftools, FV, Profit, Xanadu, Xspec, Xronos, Ximage) and DS9, Sextractor, Tempo2, IRAF, Topcat, Aladin, ROOT, EUTelescope, Geant4, IDL, GammaLib and Ctools [7].

Access to ACID is available not only via a PC but also through mobile devices (tablets or smartphones). We provide cloud data services to share and synchronize workflow outcomes, or results of ACID interactive sections, between a user's desktop and the science gateway.

The rest of the paper is organized as follows. Section 2 is an overview of relevant work carried out by other projects. Section 3 introduces the architectural design of our science gateway and discusses the implementation of the SSO technology. Section 4 describes quality assurance strategies driving the development of our gateway. Section 5 focuses specifically on ACID functionality. Session 6 presents collaborative features including visual analytics tools and data management facilities. Finally, Section 7 summarizes our work and includes pointers to future developments.

2 Background Work

The computing organizations TeraGrid and XSEDE [?] support gateways in sending jobs to HPC systems. On the other hand a number of generic frameworks have been established recently for implementing science gateways, e.g. GENIUS[1], and the outcomes of the EU-DAT EU FP7 project[4]. As a result of these developments, some domain specific gateways based on such technologies have been developed, e.g. for astronomy and astrophysics community to explore and analyse large scale datasets on DCIs. Most of these datasets are federated within the Virtual Observatory (VO) [5]. As an example, the Asteroseismic Modeling Portal [6] provides a web-based interface tied to computing resources for astronomers that use a stellar evolution code coupled with a parallel genetic algorithm to derive the properties of Sun-like stars from observations of their pulsation frequencies. The GENIUS (or Grid Enabled web eNvironment for site Independent User job Submission) grid portal is specialized for grid resources including any required security issues [7]. InSilicoLab[8] is an application portal designed to support in silico experiments by running computational software on grids. Unlike manual job submission or grid portals, InSilicoLab enables its users to run complex computations without technical knowledge of how to operate the underlyng grid resources. Born to support several chemistry applications, InSilicoLab currently supports Physics collaborations in CTA and a prototype of a science gateway[9] has been developed to optimize the resource usage on the grid with the DIRAC (or Distributed Infrastructure with Remote Agent Control) system [8].

None of the aforementioned gateway technologies has been adopted as widely as WS-PGRADE/gUSE which has more than 100 deployments [10] world-wide including over 15.000 downloads from 75 countries on its sourceforge site [11]. This framework is robust and highly sustainable as a number of on-going EU-funded projects not only support it but also continue its development, e.g. CloudSME[12], VIALACTEA[13]. Customized functionalities can be provided by the developers [14] according to the sourceforge community needs.

Fig. 1: CTA Science Gateway Architecture

WS-PGRADE/gUSE is currently being developed within the EU funded project SCI-BUS [15]. A large number of gUSE scientific gateways has been developed in the last years allowing a de-facto multidisciplinary research environment through a collaborative and community oriented technology allowing developers and end users to share layered and parameter sweep enabled workflows, workflow graphs, workflow templates, and ready-to-run workflow applications via repositories. Domain specific gUse gateway were developed in the following communities: Astrophysics [19], Life Sciences [?,9,10], Heliophysics [11,12], Materials Science [13], Earth Science [?], Bioinformatics [?] and Condensed Matter Physics[?].

WS-PGRADE/gUSE is natively compatible with SHIWA technology and its coarse-grained interoperability (CGI) approach [?] allows a de-facto workflow interoperability towards Galaxy [?], Kepler [?] and MOTEUR [?] workflow systems.

The technologies described in this section provide the background for the construction of the CTA gateway as an application-specific gateway including a wide range of functionalities as prescribed by the CTA community. A number of technological aspects are exploited e.g. WS-PGRADE/gUSE, Shibboleth, VNC.

3 Architectural Design

The architectural design of our science gateway is shown in Figure ??. This section mainly focuses on Shibboleth SSO (Section 3.2) and authentication in WS-PGRADE (Section 3.3) as these two aspects are an evolution beyond the state of the art for the WS-PGRADE technology. The cloud access (Section ??) represents an important aspect of this science gateway in comparison with the current science gateway technologies (Section 2). As regards the description of WS-PGRADE, gUSE and in particular the DCI Bridge components which is based on the standard OGSA BES[?], see [?].

ACID (Section 5) is a novel technology designed and implemented for the CTA science gateway.

3.1 Cloud

WS-PGRADE/gUSE allows direct cloud access without requiring use of any brokering platform for job submission. Any clouds that implement the Amazon EC2

interface can be accessed. If a user is already in possession of an account to access the cloud, this can be used for authentication. The cloud access process in our CTA gateway is as follows:

1. The DCI bridge administrator downloads a public base image containing a properly configured DCI bridge (this will play the role of a slave DCI bridge) from a corresponding repository. This image is saved in a target cloud environment provided that the used cloud service contains an Amazon EC2 frontend;
2. the DCI bridge administrator configures the master DCI bridge and
3. finally, the WS-PGRADE user requests an account from the cloud provider. From this point onwards the user can deploy the portal for job submission to the cloud.

3.2 Single Sign On

Security mechanisms are needed to control access to our scientific gateway and to administer computing/storage systems and services. Such mechanisms can also provide the appropriate level of protection against unauthorized access. The procedures we implemented ensure that only legitimate users can access a particular resource e.g. an archive, a computational resource, a data-set or a specific portal functionality. The differentiation between legitimate and non-legitimate users is made according to a prescribed list of attributes. This approach can also keep track of all events related to authorization and authentication and is mandatory for real-time monitoring of resource usage and for implementing an accounting mechanism.

Shibboleth is probably the most popular among the distributed solutions and offers the possibility of extending our gateway into a federation in future developments, an important feature indeed for a large community of users such as CTA. We have deployed Shibboleth which is an implementation of SAML 2.0 (or Security Assertion Markup Language), an XML-based open standard data format for exchanging authentication and authorization data between parties. This standard has been developed by the security services technical committee of OASIS (or Organization for the Advancement of Structured Information Standards), as a framework allowing exchange of assertions regarding the identity, attributes, and entitlements of a subject. Three roles are defined as follows:

Fig. 2: SAML 2.0 Web Browser SSO Profile in CTA Science Gateway

- the Principal (in our case this is a user running an HTTP user agent),
- the Identity Provider (IdP) and
- the Service Provider (SP)

The exchanging of assertions occurs between the Principal, the authority (IdP) and the consumer (SP). There are three type of assertions:

- authentication: this assertion relates to a given subject's identity;
- attribute: the subject is described by means of its attributes;
- authorization decision: this is the request to grant or deny the subject access to the specified resource.

In our CTA Science Gateway we implemented a Web Browser Single Sign On Profile using Shibboleth 2.0. The Authentication process in the CTA Science Gateway is as follows (Fig. 1) :

1. The HTTP user agent (browser) requests the target resource (CTA Science Gateway access) at the service provider. The service provider checks if a valid security context already exists for the target resource, if so it provides the access (point 8).
2. If not the service provider redirects the user agent to the SSO Service at the identity provider.
3. The user agent issues a GET request to the SSO Service at the IdP providing a SAMLRequest. The IdP processes the request and performs a security check. If the user does not have a valid security context, the identity provider identifies the user.
4. It responds to the SAMLRequest with a document containing an XHTML form.
5. The user agent issues a POST request to the service provider (assertion consumer service) providing the SAMLResponse from the XHTML form (point 4)
6. A security context is created at Service Provider level by the assertion consumer service and the user agent is redirected to the the target resource (CTA Science Gateway access).
7. Like in point 1) the HTTP user agent (browser) requests the target resource (CTA Science Gateway access) at the service provider.
8. The security context is now created and the service provider returns the access to the Science Gateway.

3.3 Authorization in WS-PGRADE

In our architectural design of the infrastructure Tomcat application server runs the Scientific Gateway. Tomcat

Fig. 3: The AJP protocol is used for communication between Tomcat and Apache

Fig. 4: Using this architecture Apache can perform Load Balancing

is integrated with Apache by using Apache JServ Protocol (AJP). Apache receives the incoming request that is forwarded via AJP to Tomcat and the response will also be sent back to the Apache server via AJP (Fig. 2). This last scenario facilitare load-balancing and thus provides a most generic and scalable solution fitting the requirements of a large community (Fig. 3). In our architecture Apache will intercept requests, and after performing all authentication related tasks, it will pass these to the backend servlet container using AJP. After the authentication the attributes describing the user are passed to the Scientific Gateway. A particular Liferay hook plugin [16] will then perform the following:

- auto creation of a new user using Shibboleth attributes (email, first name, last name),
- auto update of user information upon login and
- role mapping.

The list of attributes describing a given user can be used at application level in order to perform a detailed authorization for accessing archives, applications and remote resources.

4 Testing

The overall quality of the CTA gateway services is governed by a well defined quality assurance strategy. Verification and testing software packages as well as Agile software development practices have been employed. This section outlines the particular metrics used during the development. Software quality is measured by a number of metrics such as the number of lines of code, the cyclomatic complexity (related to the complexity of a code) or the coverage metrics (measuring the degree to which the code is tested by a test suite). Static analysers, e.g. FindBugs[17] for portlets, have been used to analyse software code looking for possible bugs.
Furthermore web testing (e.g. interface functionality and browser compatibility) can give good insights into

---

[16] `https://github.com/ivan-novakov/liferay-shibboleth-plugin`
[17] `http://findbugs.sourceforge.net/`

several important aspects such as TCP/IP communication, firewall rules, applications running in web pages or on the server side, security and performance implications. Regarding the functional testing of web applications, the validity of the links in web pages, and on the input forms used for getting information from the user should be checked. Data integrity and errors during edit, delete, modify, insert and any DB-related functionality can be verified in our implementation. Concerning interface testing in the case of web applications, mainly the communication between web server, application server, database server, external web services are tested. The aforementioned tests are performed employing a number of testing tools including: Robot framework[18] (generic test automation environment) with Selenium[19] (web test automation tool) and Sikuli[20] (image based GUI test tool). Robot framework has been used because it is a platform and application independent way of testing. It can be used for almost every type of acceptance and system testing and it enables high-level syntax for creating test cases in a uniform way.

A number of performance tests have been done [?] to check the scalability of the DCI Bridge component and the load balance algorithm used in cloud environment. Further performance results will be produced when the cloud environment (Section 7) will allow to provide an intelligent workload management platform for the ACID services .

5 ACID: the Astronomical and Physics Cloud Interactive Desktop

The underlying idea for the Astronomical & Physics Cloud Interactive Desktop [16], as employed in the CTA Science Gateway, is the use of many software packages without any installation on a local desktop. The users can exploit the native GUI of the packages that are included in ACID. For using interactively remote programs, ACID exploits an ad hoc VNC-based User Interface (VUI). Web-based User Interfaces (WUI) often do not offer a full control or an interactive mode like specific native GUIs. VNC-based User Interface is a good robust and popular solution for environments that pack together a number of heterogeneous programs that can be accessed by using their native GUIs. VUI is independent from the operating system. Moreover ACID does not require any VNC client installed on the user desktop and it is also able to ensure the network access security. The VNC-based User Interfaces run on the ACID server

and allow to use any GUI of the programs launched remotely. ACID uses *ownCloud*[21] to share intuitively data between the user desktop and ACID server.

Data sharing with the ACID server is achieved using the *Unison File Synchronizer*[22]. Each desktop client (running the *ownCloud* client) synchronizes and shares a user directory with the portal through the ACID server. ownClient clients are publicly available for Windows (XP, Vista, 7 and 8), Mac (OS X 10.7+, 64bit) Linux (CentOS, Debian, Fedora, openSUSE, SLE, Ubuntu) maintained by OwnClient developer community. A web-based access is provided as well enabling the file management via browser.

The astronomical and physics software suites available in ACID[23] are: *Heasoft (High Energy Astrophysics Science Archive Research Center)* that contains more than 100 subprograms like *Ftools*, (General FITS file utility programs and mission-specific data analysis tools), *FV* (Interactive editor and viewer for astronomical data files in FITS format), *Profit* (for Visualizing and Modelling High-Resolution Spectra), *Xanadu* (Data Analysis for X-Ray Astronomy), *Xspec* (X-Ray Spectral Fitting Package), *Xronos* (Timing Analysis Software Package), *Ximage* (X-Ray Image Package) and then *DS9* (Astronomical imaging and data visualization application), *Sextractor* (builds catalogue of objects from an astronomical image), *Tempo2* (Pulsar timing package), *IRAF* (Image Reduction and Analysis Facility) and *IRAF Ximtool* (X11IRAF Tools), *Topcat* (Tool for Operations on Catalogue and Tables), *Aladin* (Interactive software for sky atlas), *ROOT* (Set of Object Oriented frameworks to handle and analyze large amounts of data), *EUTelescope* (Generic Pixel Telescope Data Analysis Framework), *Geant4* (Toolkit for simulation of the passage of particles through matter), *VisIVO* (Tools for image and movie generation from large scale dataset).

The ACID environment can be exploited in two ways: as a Distributed Computing Infrastructure (to perform an off-line analysis through workflow submission) or interactively (to perform an on-line analysis via the Interactive Desktop).

Off-line analysis (Fig. 5) allows submitting jobs to ACID, as it acts as a node of a PBS cluster. The job is submitted as a workflow and it is possible to exploit the large number of command line-based software packages available in the environment. At the time of writing there are over 150 software packages included in ACID that occupy over 5GB, and cover a number of tasks

---

[18] http://code.google.com/p/robotframework
[19] http://seleniumhq.org
[20] http://sikuli.org

[21] https://owncloud.org
[22] http://www.cis.upenn.edu/~bcpierce/unison/
[23] https://acid.oact.inaf.it/ACID/Included_packages.html

typically performed by astronomers, from visualization to data analysis. The overall integration of the aforementioned tools within ACID is maintained (through regular updates), including extensive documentation, tutorials and examples. For each new stable software release, a testing phase takes place prior to publication on ACID.

The aforementioned packages do not occupy space on the user's machine as they live in one or more virtual desktop environments depending upon the deployment followed (i.e. they are not placed inside the gateway). Prospective users are encouraged to submit questions to the developers with any possible problems identified.

The on-line analysis functionality (Fig. 4) allows to fully exploit the potential of the platform. Once the user gets access to the portal through the Shibboleth Single Sign On procedure the ACID portlet (Fig. 6) manages the creation of the Interactive Environment for a web browser or for a mobile device. Currently we support iOs devices fully, but an Android version is planned for future developments. As a first step the portlet analyzes the User Agent string: a piece of data transmitted in the HTTP header during the web request, containing information about the browser type and system information. A pattern recognition alghorithm identifies an iPad user and provides the security environment to access the Remote Desktop, then, using an IPhone URL Schema[24], the VNC client [25] is initiated and connected to the Remote Desktop. If no mobile web browser is detected the client applet launches the ACID User Interface.

Once the ACID portlet is executed for the first time:

- a linux user account is created on the ACID platform for each authenticated user. The UID (Unique IDentifier) Shibboleth attribute is used as a login name.
- the OwnCloud credentials are activated for the user on the ACID-OwnCloud server[26] and
- the user is notified by e-mail with the parameters needed for the mobile connection.

6 Collaborative Communities

The CTA Science Gateway offers a collaborative and a community oriented environment due to the underlying workflow system paradigm and the usage of the Liferay and WS-PGRADE/gUSE [4][3][17] collaboration and sharing facilities. The WS-PGRADE/gUSE framework contains internal repositories, which allow researchers

of the community, to publish and share artefacts. Different artefacts, for example applications, portlets, workflows, etc. that could be specific to a particular application domain or science gateway could also be efficiently shared and reused between multiple developer and research communities. Sharing these various artefacts in repositories, for example in application, workflow or portlet repositories, enable gateway developers to reuse them when building a new gateway and/or creating a new application. Moreover, these building blocks can in many cases be utilized by developers using different gateway frameworks and in this way facilitate an even wider collaboration between communities. Sharing and reuse of these artefacts via repositories significantly shortens the development time. To support inter- or multidisciplinary research cooperation, the gUSE framework is connected to external repositories containing portlets (the SCI-BUS Portlet Repository[27]) and workflows (the SHIWA Workflow Repository[28]). Developers can upload and publish portlets and workflows in these repositories. Both developers and researchers of different communities can access these repositories, find and download those portlets and workflows they want to use. To facilitate exchange between different groups, in particular enabling sharing and realizing re-use and interoperability between different workflow systems and DCI resources, the SHIWA (SHaring Interoperable Workflows for large-scale scientific simulations on Available DCIs) Simulation Platform can be employed. It provides solutions to facilitate sharing and exchanging of workflows through the following tools:

- SHIWA Repository: A database where workflows and meta-data about workflows can be stored. The database is a central repository for users to discover and share workflows within and across their communities.
- SHIWA Portal[29]: A web portal that is integrated with the SHIWA Repository and includes a workflow executor engine that can orchestrate various types of workflows on a number of computational grid/cloud platforms.

Through the SHIWA Portal one can define and run simulations on the SHIWA Virtual Organisation which is an e-infrastructure that gathers computing and data resources from various DCIs, including the European Grid Infrastructure[30]. The portal (via third party workflow engines) provides support for a number of commonly used workflow engines and it can be extended

---

[24] http://goo.gl/aiHvKe
[25] http://goo.gl/vWmxhq
[26] https://acid.oact.inaf.it/owncloud/

[27] https://scibus-repo.cpc.wmin.ac.uk
[28] http://shiwa-repo.cpc.wmin.ac.uk
[29] http://shiwa-portal2.cpc.wmin.ac.uk/
liferay-portal-6.1.0
[30] http://www.egi.eu

Fig. 5: ACID as a user interface for on-line analysis

Fig. 6: ACID as off-line analysis tool

Fig. 7: ACID Liferay portlet flow diagram

with other engines on demand. Such extensions translate between workflow languages and facilitate the nesting of workflows into larger workflows even when those are written in different languages and require different interpreters for execution. This functionality can enable scientific collaborations to share and offer workflows for reuse and execution. Shared workflows can be executed on-line, without installing any special client environment for downloading workflows. Several shared workflows are furnished with application specific portlets to provide input parameters and to display the workflow results in a user-friendly way. Those portlets are made available to the public on the SCI-BUS Portlet Repository. This repository is an exciting hub for sharing, browsing and downloading Liferay portlets developed within the SCI-BUS project. It leverages on the entire SCI-BUS ecosystem to release and share portlets in a user-friendly, one-stop site. Scientific communities using Liferay based gateways can take benefit of downloading portlets from this portlet repository. Each application specific portlet contains the details of the related underlying workflows pointing to the specific SHIWA workflow repository entries. Thanks to these facilities, the CTA Science Gateway has been endowed with astrophysical portlets and workflows devoted to dataset exploration and scientific visualization as described in Section 6.1 and tools for the management of input datasets and their processing outcomes reported in Section 6.2. Some of the portlets required for CTA have been developed starting from the portlets available on the SCI-BUS portlet repository, e.g. Section 6.1 and 6.2. However new portlets have also been developed to specifically address the needs of the CTA community. e.g. ACID.

6.1 Visual data exploration

Visual data exploration is central to scientific discovery [18]. Most analysis tasks require some kind of visualizations to be created and explored off-line or interactively. Visual exploration and discovery tools [19, 15] are therefore invaluable instruments which could provide prompt and intuitive insights into the intrinsic data characteristics, enabling scientists both to rapidly identify interesting areas within which to apply computationally expensive algorithms and also to discover correlations in data patterns.

Our scientific gateway provides an integrated methodology for addressing the visual exploration of big astrophysical datasets. The rendering process is typically computationally intensive, often requiring millions of floating-point and integer operations for generating single 3D views. The visualization engine of this scientific gateway is a full C++ implementation: VisIVO [20,21], a software framework for effective visual discovery in large-scale astrophysical data sets [22,23,24]. VisIVO is the only package included in the CTA portal that implements fully its functionalities by means of WS-PGRADE/gUse workflows and portals. The standard visualization views provided by this package employ orthographic projections. The system supports several algorithms based on the Visualization Toolkit [?] and Splotch [25,26]. In the latter visualization algorithm, a customized ray tracer [27], is employed for effectively creating meaningful 3D views large datasets. The intensive underlying calculation is optimized by a preprocessing phase, and recent developments [28] exploit GPUs for hardware acceleration.

A number of customized workflows [19,29] have been downloaded from the SHIWA repository and are configured to be available by default to CTA users, e.g. to allow the upload of datasets and creation of scientific movies. These workflows are provided with specific user interface Liferay portlets[31] (available on the SCI-BUS portlets repository) to enable easy parameter setting for standard users hiding the complexity of the underlying system and infrastructures. The creation of a movie requires often hundreds or thousands high quality images must be produced. For this reason Parameter Sweep (PS) workflows are employed. PS workflows [30] (which are supported by the WS-PGRADE/gUSE infrastructure) are executed in distributed parallel computations.

6.2 The File System and its Data Model

A Data Management portlet [19] has been employed to allow users to share artefacts (e.g. workflows) within

---

[31] http://goo.gl/oOpc1F

Fig. 8: Data model DB

the gateway for managing their datasets and workflow outcomes. The underlying data-model is implemented through a MySql RDBMS. The metadata associated with the uploaded datasets as well as the produced results are encapsulated by customized *GenericFile* objects. These objects contain attributes such as the *owner* or the *creation_time*. The specific objects are defined by inheriting from the GenericFile introducing new attributes such as endianity, number of fields, type and number of elements in the case of a tabular dataset *TableData*. Each specific object can contain one or more fields table including e.g. details and statistics on each field of the tabular dataset. Also the metadata of the visualization results produced with the tools described in Section 6.1 are described as *Image* objects. See Figure 7 for more details on the database structure. Additionally a WS-PGRADE tool named Data Avenue[**?**] provides a useful file commander for data transfer, enabling easy data moving between various storages services (such as grid, cloud, cluster, supercomputers) by various protocols (HTTP, HTTPS, SFTP, GSFTP, SRM). The usage of Data Avenue is similar to other well-known file commanders (e.g. Total Commander[32] or Midnight Commander[33]): users can interact with a simple two-panel (source and destination panel) to copy, delete, download or upload remote files.

7 Conclusions and Future Work

We used a set of wide adopted standards and tools such as Java Portlet Specification, SAML 2.0, Shibboleth 2.0, WS-PGRADE/gUSE, with the aim of enlarging the developer community and improving the sustainability of the software project. This solution provides an ecosystem of new technologies: mobile access, federated authentication, workflow engine and cloud services with an high level of flexibility in order to tailor a product that suits the present and future requirements of the CTA community. Our solution provides an ecosystem of new technologies (mobile access, federated authentication, workflow engine and cloud services) with a high level of flexibility facilitating customization to suit not only present but also future requirements of the CTA community.

We discussed background work on science gateways and presented details of the architectural design of our solution highlighting its relevant benefits. Finally several mechanisms facilitating collaborative activities have been discussed focusing on visual data exploration.

We will also look into gathering user experiences once the gateway is operational to improve functionalities and facilitate planning future developments The future development of the CTA Science Gateway will be driven by the present and forthcoming user requirements of CTA community. We are integrating the OpenNebula cluster[34] hosted by the Astrophysical Observatory of Catania (96 CPU cores and 1024GB RAM). This cloud environment will allow to provide an intelligent workload management platform for the ACID services towards overcoming potential bottlenecks in accessing these services.

References

1. A. Andronico, R. Barbera, A. Falzone, P. Kunszt, G. Lo Re, A. Pulvirenti, and A. Rodolico, "Genius: a simple and easy way to access computational and data grids," *Future Generation Computer Systems*, vol. 19, no. 6, pp. 805–813, 2003.
2. W. Schroeder, K. Martin, L.S. Avila, C.C. Law "The Visualization Toolkit User's Guide, Version 4.0," in *Kitware, version 4* 2001
3. J. Kocot, T. Szepieniec, D. Harezlak, K. Noga, and M. Sterzel, "Insilicolab–managing complexity of chemistry computations," in *Building a National Distributed e-Infrastructure–PL-Grid.* Springer, 2012, pp. 265–275.
4. M. Riedel, P. Wittenburg, J. Reetz and others, "A data infrastructure reference model with applications: towards realization of a ScienceTube vision with a data replication service," in *Journal of Internet Services and Applications*, vol. 4, no. 1, pp. 1–17, 2013.
5. J. Towns, T. Cockerill, M. Dahan, I. Foster and others, "XSEDE: accelerating scientific discovery," in *Journal of Internet Services and Applications*,vol. 16, no. 5, pp. 62–74, 2014.

---

[32] http://www.ghisler.com/
[33] http://www.midnight-commander.org/

[34] opennebula.org/
[35] http://www.sci-bus.eu/

6. G. Terstyanszky, T. Kukla, T. Kiss, S. Winter, P. Kacsuk, A. Balasko "Sharing Workflows through Coarse-Grained Workflow Interoperability," in *GRID'2012, 5th International Conference on Distributed Computing and Grid Technologies in Science and Education, Dubna, Russia, 17-21 July 2012.*

7. J. Goecks, A. Nekrutenko, T. Anton, J. Taylor "Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences," in *Genome Biol*, vol. 11, no. 8, pp. R86, 2010.

8. B. Ludscher, I. Altintas, C. Berkley and others, "Scientific workflow management and the Kepler system," in *Concurrency and Computation: Practice and Experience*, vol. 18, no. 10, pp. 1039–1065, 2006.

9. T. Glatard, J. Montagnat, D. Lingrand and others, "Flexible and efficient workflow deployment of data-intensive applications on grids with moteur," in *International Journal of High Performance Computing Applications*, vol. 22, no. 3, pp. 347–360, 2008.

10. P. Kacsuk, Z. Farkas, M. Kozlovszky, G. Hermann, A. Balasko, K. Karoczkai, and I. Marton, "Wspgrade/guse generic dci gateway framework for a large variety of user communities," *Computing in Science & Engineering*, vol. 10, no. 4, pp. 601–630, 2012.

11. P. Kacsuk, J. Kovacs, Z. Farkas, A. C. Marosi, and Z. Balaton, "Towards a powerful european dci based on desktop grids," *Journal of Grid Computing*, vol. 9, no. 2, pp. 219–239, 2011.

12. M. Marzolla and others "Open standards-based interoperability of job submission and management interfaces across the grid middleware platforms glite and unicore," in *e-Science and Grid Computing, IEEE International Conference*, vol. 1, 2007, pp. 592–601.

13. Péter Kacsuk (ed.), "Science Gateways for Distributed Computing Infrastructures: Development Framework and Exploitation by Scientific User Communities," Springer, 2014. p. 301. (ISBN:978-3-319-11267-1)

14. S. Gesing and others., "A single sign-on infrastructure for science gateways on a use case for structural bioinformatics" *Journal of Grid Computing*, vol. 10, no. 4, pp. 769–790, 2012.

15. K. Borne, "Virtual observatories, data mining, and astroinformatics," in *Planets, Stars and Stellar Systems*. Springer, 2013, pp. 403–443.

16. M. Woitaszek, T. Metcalfe, and I. Shorrock, "Amp: a science-driven web-based application for the teragrid," in *Proceedings of the 5th Grid Computing Environments Workshop*, vol. 1, 2009, pp. 1–7.

17. R. Barbera, G. Andronico, G. Donvito, A. Falzone, J. Keijser, G. L. Rocca, L. Milanesi, G. P. Maggi, and S. Vicario, "A grid portal with robot certificates for bioinformatics phylogenetic analyses," *Concurrency and Computation: Practice and Experience*, vol. 23, no. 3, pp. 246–255, 2011.

18. A. Tsaregorodtsev, M. Bargiotti, N. Brook, A. C. Ramo, G. Castellani, P. Charpentier, C. Cioffi, J. Closier, R. G. Diaz, G. Kuznetsov *et al.*, "Dirac: a community grid solution," in *Journal of Physics: Conference Series*, vol. 119, no. 6. IOP Publishing, 2008, p. 062048.

19. S. Shahand, A. Benabdelkader, M. M. Jaghoori, M. a. Mourabit, J. Huguet, M. W. Caan, A. H. Kampen, and S. D. Olabarriaga, "A data-centric neuroscience gateway: design, implementation, and experiences," *Concurrency and Computation: Practice and Experience*, 2014.

20. M. Jaghoori, A. van Altena, B. Bleijlevens, and S. Olabarriaga, "A grid-enabled virtual screening gateway," in *Proceedings of the 6th International Workshop on Science Gateways*, June 2014.

21. G. Pierantoni, B. Coghlan, E. Kenny, P. Gallagher, and D. Perez-Suarez, "Extending the sheba propagation model to reduce parameter-related uncertainties," *Computer Science*, vol. 14(2), pp. 253–272, 2012.

22. G. Pierantoni and E. Carley, "Metaworkflows and workflow interoperability for heliophysics," *IEEE Xplore Digital library*, 2014.

23. Y. Gordienko, L. Bekenov, O. Gatsenko, E. Zasimchuk, and V. Tatarenko, "Complex workflow management and integration of distributed computing resources by science gateway portal for molecular dynamics simulations in materials science," *Proc. 3rd International Conference on High Performance Computing (HPC-UA 2013), October 8-10, 2013, Kyiv, Ukraine*, 2014.

24. A. Costa, U. Becciani, P. Miocchi, V. Antonuccio, R. Capuzzo Dolcetta, P. Di Matteo, and V. Rosato, "Astrocomp: web technologies for high performance computing on a network of supercomputers," *Computer physics communications*, vol. 166, no. 1, pp. 17–25, 2005.

25. A. Costa, U. Becciani, P. Massimino, M. Krokos, G. Caniglia, C. Gheller, A. Grillo, and F. Vitello, "Visivoweb: a www environment for large-scale astrophysical visualization," *Publications of the Astronomical Society of the Pacific*, vol. 123, no. 902, pp. 503–513, 2011.

26. P. Massimino, A. Costa, U. Becciani, F. Vitello, and E. Sciacca, "Acid: an interactive desktop for cta science gateway," in *Proceedings of the 6th International Workshop on Science Gateways*, June 2014.

27. A. Balasko, Z. Farkas, and P. Kacsuk, "Building science gateways by utilizing the generic ws-pgrade/guse workflow system," *Computer Science*, vol. 14, no. 2, 2013.

28. A. Hassan and C. J. Fluke, "Scientific visualization in astronomy: Towards the petascale astronomy era," *Publications of the Astronomical Society of Australia*, vol. 28, no. 02, pp. 150–170, 2011.

29. E. Sciacca, M. Bandieramonte, U. Becciani, A. Costa, M. Krokos, P. Massimino, C. Petta, C. Pistagna, S. Riggi, and F. Vitello, "Visivo workflow-oriented science gateway for astrophysical visualization," in *Parallel, Distributed and Network-Based Processing (PDP), 2013 21st Euromicro International Conference on.* IEEE, 2013, pp. 164–171.

30. U. Becciani, A. Costa, V. Antonuccio-Delogu, G. Caniglia, M. Comparato, C. Gheller, Z. Jin, M. Krokos, and P. Massimino, "Visivo–integrated tools and services for large-scale astrophysical visualization," *Publications of the Astronomical Society of the Pacific*, vol. 122, no. 887, pp. 119–130, 2010.

31. A. Costa, P. Manzato, U. Becciani, M. Comparato, V. Costa, F. Gasparo, C. Gheller, A. Grillo, M. Molinaro, F. Pasian *et al.*, "The tvo archive for cosmological simulations: Web services and architecture," *Publications of the Astronomical Society of the Pacific*, vol. 120, no. 870, pp. 933–944, 2008.

32. G. Caniglia, M. Krokos, U. Becciani, C. Gheller, R. Nichol, M. Comparato, A. Costa, A. Grillo, Z. Jin, P. Massimino *et al.*, "Visual discovery in large-scale astrophysical datasets; experiences using the sloan digital sky survey," in *Visualisation, 2009. VIZ'09. Second International Conference in.* IEEE, 2009, pp. 10–15.

33. U. Becciani, V. Antonuccio-Delogu, A. Costa, and C. Petta, "Cosmological simulations and data exploration: a testcase on the usage of grid infrastructure," *Journal of Grid Computing*, vol. 10, no. 2, pp. 265–277, 2012.

34. S. Riggi, V. Antonuccio, M. Bandieramonte, U. Becciani, F. Belluomo, M. Belluso, S. Billotta, G. Bonanno, B. Carbone, A. Costa *et al.*, "A large area cosmic ray detector for the inspection of hidden high-z materials inside containers," in *Journal of Physics: Conference Series*, vol. 409, no. 1.   IOP Publishing, 2013, p. 012046.

35. K. Dolag, M. Reinecke, C. Gheller, and S. Imboden, "Splotch: visualizing cosmological simulations," *New Journal of Physics*, vol. 10, no. 12, p. 125006, 2008.

36. Z. Jin, M. Krokos, M. Rivi, C. Gheller, K. Dolag, and M. Reinecke, "High-performance astrophysical visualization using splotch," *Procedia Computer Science*, vol. 1, no. 1, pp. 1775–1784, 2010.

37. M. Levoy, "Efficient ray tracing of volume data," *ACM Transactions on Graphics (TOG)*, vol. 9, no. 3, pp. 245–261, 1990.

38. M. Rivi, C. Gheller, M. Krokos, K. Dolag, and M. Reinecke, "Gpu accelerated particle visualization with splotch," *arXiv preprint arXiv:1309.1114*, 2013.

39. E. Sciacca, M. Bandieramonte, U. Becciani, A. Costa, M. Krokos, P. Massimino, C. Petta, C. Pistagna, S. Riggi, and F. Vitello, "Visivo science gateway: a collaborative environment for the astrophysics community," in *Proceedings of the 5th International Workshop on Science Gateways*, 2013.

40. F. Chirigati, V. Silva, E. Ogasawara, D. de Oliveira, J. Dias, F. Porto, P. Valduriez, and M. Mattoso, "Evaluating parameter sweep workflows in high performance computing," in *Proceedings of the 1st ACM SIGMOD Workshop on Scalable Workflow Execution Engines and Technologies.*   ACM, 2012, p. 2.

41. Ákos Hajnal, Zoltán Farkas, Péter Kacsuk, Támas Pintér "Remote Storage Resource Management in WS-PGRADE/gUSE" In: Péter Kacsuk (ed.), "Science Gateways for Distributed Computing Infrastructures: Development Framework and Exploitation by Scientific User Communities," Springer, 2014. pp. 69-81. (ISBN:978-3-319-11267-1)