The Grid as a Software Application Provider in a Synchrotron Radiation Facility

Roberto Pugliese, Milan Prica, George Kourousias, Andrea Del Linz, and Alessio Curri

Abstract In this chapter we introduce the concept of the Grid as an *Application Service Provider*. We discuss the implementation and deployment of a fully operational system in a Synchrotron Radiation Facility. The Grid is traditionally used for HTC computations and not often as a provider for Software on Demand; especially for non-Grid specific GUI based applications. As a computing paradigm, the Grid is successfully utilised by the experimental scientific community. Due to the data-parallelism of the computational problems in the general field of physical sciences and the very high data volumes in terms of storage, the Grid has been closely related to High Throughput and High Performance Computing (HTC/HPC). A substantial number of software applications in the form of scientific computing utilities that do not require HPC are used in Synchrotron labs and other similar establishments. We envisioned a common web portal where the lab user can access a defined set of pre-configured *scientific computing* applications[1] that rely on Grid resources. The pilot application has been in the SRF Elettra.[2] The model and the intermediate subsystems have been developed and suitably configured. In this chapter we propose (1) a "Software as a Service" model (SaaS) as an alternative deployment for these applications, (2) we introduce an infrastructure where our SaaS model is fully integrated to the Grid – utilising remote computation and storage resources, (3) we outline our technology layout that enables interactivity, transparent access to storage, Web Grid portals, virtual collaboration environments and remote access to SRF instrumentation.

M. Prica (✉)
Scientific Computing Group, Information Technology Department, ELETTRA Sincrotrone Trieste, Trieste, Italy
e-mail: milan.prica@elettra.trieste.it

[1] the Scientific Computing in the domain of an SRF often deals with mathematical models for the analysis of the collected data. Typical applications are those of Imaging.

[2] Elettra – Sincrotrone Trieste S.C.p.A.. URL: http://www.elettra.trieste.it

F. Davoli et al. (eds.), *Remote Instrumentation Services on the e-Infrastructure*,
DOI 10.1007/978-1-4419-5574-6_3, © Springer Science+Business Media, LLC 2011

1 The Classical Grid In An SRF

The modern High Energy Physics (HEP) establishments like the Synchrotron Radiation Facilities (SRF) have extended and specialised computing requirements. Approaches that offer High Performance Computing (HPC) and High Throughput Computing (HTC) have long been utilised in order to process, analyse, and archive the vast amount of the generated experimental data [1, 2]. Advanced computing paradigms like those of Parallel computing and Grid [3], are considered the major approach for computing in such facilities. The former, Parallel computing, is a mature solution that is well suited since the computational problems in HEP tend to be data-parallel. The latter, the Grid Computing paradigm, is less widespread but has proved its potential [4] and is widely used in major setups [5]. A major characteristic of the Grid is that it can include parallel computing infrastructures and often it blends as a superstructure for HTC, HPC and generic distributed computing. In this chapter we focus on a different and overlooked class of computational needs that may appear in Synchrotron Radiation Facilities but are not related to HTC [6].

2 Non-HTC Scientific Software For SRF Labs

Elettra and other SRFs host a number of beamlines and their corresponding end-stations and laboratories. These labs are hosting external users who perform experiments with the assistance of the scientific personnel of the facility. Due to the non-standard nature of the specialised instrumentation, the initial data analysis and processing is based on established workflows that do not necessarily require HPC. Such workflows may include the use of various software applications for tasks such as opening a custom data format, applying an instrument-specific filter, converting imaging data, calibrating, searching past data and configurations. The in-house scientists may train their hosted users to perform on their own the workflow and its sequential steps. In most of the labs, the process requires that the external users utilise an internal computer/terminal with pre-installed software or reinstall the required software in their personal computers if the licensing allows. This way of software deployment may lead to a series of inconveniences including installation problems, incompatibilities, variations in software configurations, unexpected processing times due to diversity of hardware, versioning issues, data redundancy, and other.

3 Software As A Service, On Grid

Our solution suggests the consideration of an alternative software deployment method, that of Software as a Service (SaaS) [7, 8] that can be viewed in its general form as On-demand software. Additionally we suggest the utilisation of the existing Grid infrastructure to act as the required Application Service Provider (ASP). Such a Vertical Market [9] solution should enable any authorised beamline user to access the collection of software applications necessary for the lab's workflow and use them in a unified Web-based environment. The SaaS system we propose is suitably blended in the Grid so the computation, storage and sharing of the data is always remote. This yields the obvious advantage of utilising remote and potentially more powerful hardware than that of the user's personal computer. During the design phase the HTC orientation of the current Grid infrastructure was critically examined. We concluded that there are many software applications in the SRF community that do not need HTC but could still benefit from the established Grid infrastructures. This traditional infrastructure has been deployed mainly for the computation of large batch calculations [3] and network data I/O. In our case we had to enable a user friendly SaaS solution without altering the HTC character of the existing systems. In order to enable the solution we propose: (i) a Grid Web Portal for the user, administrator, and developer, (ii) further development of the Interactive Grid, (iii) suitable extensions of the Middleware to incorporate SRF needs.

4 A Web Portal for the Grid services

As a Grid Web portal, we have developed the Virtual Control Room (VCR) [10]. Its aim is to provide the user with access to Grid resources and services through a single login from a Web browser. The VCR allows profile and role based user accounts thus one can login as beamline personnel, external user, administrator, or developer. There is an ongoing development of additional features including functionalities such as on-line collaboration and remote instrument control. The core development of the VCR is on the server-side of the system where the actions on the user-friendly web interface are suitably forwarded to the Grid middleware. In the case we propose, an external user logs-in the VCR and based on the user's profile, a predefined set of software applications may be launched. This software is not necessarily web-based, but *web-delivered* since the computation is on a remote computing resource and the VCR provides the required technologies that enable user interaction through a web browser.

5 Interactivity, Data Storage and the VCR

The general notion of interactivity on the Grid, although it has been recently progressed and advanced by successful large projects [11–13], is still a relatively new area. For our solution we aimed at enabling standard applications that assume interaction with the user. The technology we investigated is that of i2glogin. This enables an i2glogin-initiated application running on a Grid Worker Node (WN) to communicate through secure channels and interact with the server side of i2glogin that usually resides on the system that launched the application; the User Interface (gLite UI)[3]

[3] The UI [14] should not be confused with the popular term of Graphical User Interface (GUI).

that the job was submitted from. The desired results were those of remote visualisation and application steering. In our case, the user has access to the VCR and not to a traditional UI, so the server instance of i2glogin has to be integrated on the VCR server. Through suitable tunneling technologies and Desktop sharing (VNC) techniques, the VCR client in form of a web portal accessed from the user's computer allows interaction even with applications with GUI. Since the user works only with a web browser and the applications runs on a remote WN, the user can be agnostic of the application's native platform, configuration, and hardware requirements. In the process of enabling legacy desktop applications it was desired to access Storage Elements (SE) for full I/O. There are two main approaches for I/O from an application to a SE. The first is a through *synchronisation* of the WN's temporary storage space with that of a SE through scripts and tools like GridFTP [15] and UberFTP. The second approach is that of a custom designed application where the necessary mechanisms are integrated in the application so that the I/O to a SE in done through a suitable API. The problem rises in the simple case that the user needs to open or save a file by means of a legacy graphical dialog (in a GUI based application) that displays only the WN local filesystem. For solving this issue we oriented towards virtual filesystem solutions [16] that could mount locally a remote SE. Such solutions proved to be intrusive since they required an installation overhead so we adopted an alternative approach; that of Parrot [17]. Parrot enables the attachment of existing programs to remote I/O systems through the filesystem interface; in our case the remote system is GridFTP in a SE. There are no installation requirements and not classical mounting but library call intercepting so there is a trade-off in performance. The *online collaboration* [18] functions of the VCR combined with the use of a common storage in the Grid (SE) can enrich the use of the software applications since the scientists can exchange information in real-time, access common datasets, discuss configurations and redefine workflows.

6 Enabling Scientific Instruments on the Grid

An advancement especially for SRF facilities, is the enhancement of the current Grid middleware with a suitable system that enables the inclusion of scientific instrumentation and devices such as robots, sensors, cameras, motors and other instruments. The concept of this grid middleware extension is the Instrument Element (IE). We introduce an evolved and advanced version of Instrument Element for remote operation and control of instruments[19–21]. It is an addition to the existing core Grid parts; the Computing Element (CE) and the Storage Element (SE) that serve the purposes that their names suggest. The IE that we are referring to, and the related technologies, have been developed in the EU project on the Deployment of Remote Instrumentation Infrastructure (DORII).[4] In the vertical market of Synchrotron

[4] The DORII project is supported by the European Commission within the 7th Framework Programme (FP7/2007–2013) under grant agreement no. RI-213110. URL: http://www.dorii.eu

Radiation Facilities “grid-enabling” scientific devices has a very high impact. The scientific instrumentation that has been Grid-enabled can be remotely accessed and steered by authorised scientists that may be off-site. By providing a solution that enables remote instrument control through the common web portal of the VCR, the SaaS deployed application can organically connect to the source of the data. In practical terms: through the VCR the user can securely control the instrument, virtually collaborate with other scientists, collect the data, access them from distributed storage, use a set of preconfigured applications and process the data in remote computing infrastructures on the Grid.

7 Conclusions

The Grid, by principle, should be flexible, extensible and scale well. The HEP community has utilised the characteristics of its infrastructure for hight throughput computational problems. We extend this functionality by deploying a total infrastructure for enabling software on-demand services on the Grid. The SaaS class we target is that on scientific computing tasks and especially that of non-HTC software for SRF beamlines and labs. We aim at an improved end user experience and this has resulted in a common interaction environment, the VCR, with numerous services and features. Additional extensions, like the IE, on the Grid middleware have bridged the gap between software, the Grid and specialised scientific instrumentation. The pilot deployment was for the Small Angle X-ray Scattering (SAXS) beamline[5] in the SRF Elettra. There is work in progress for the further development of the above concepts in order to enable a wider range of applications and use scenarios. Areas of technical interest include that of performance issues, data visualisation, optimisations and middleware advances. We consider the Grid a viable computing paradigm for handling the computational needs of Synchrotron Radiation Facility and we believe that the introduction of additional services like SaaS can be beneficial, assuming that does not interfere with the current uses of it.

Acknowledgements We thank our partners in the DORII project for the ongoing exchange of scientific ideas. Many thanks to Herbert Rosmanith (i2glogin) and Douglas Thain (Parrot). Finally we thank the two unknown referees for their helpful comments.

This works has been partially supported by the DORII EU project (European Commission within the 7th Framework Programme (FP7/2007-2013) under grant agreement no. RI-213110).

[5] A high-flux Small Angle X-ray Scattering (SAXS) beamline mainly intended for time-resolved studies on fast structural transitions in the sub-millisecond time region in solutions and partly ordered systems with a SAXS-resolution of 1–140 nm in real-space. URL:http://www.elettra.trieste.it/experiments/beamlines/saxs/index.html

References

1. J. Basney, M. Livny, and P. Mazzanti. Harnessing the capacity of computational grids for high energy physics. Conference on computing in high energy and nuclear physics, 2000
2. D.P. Benjamin. Grid computing in the Collider Detector at Fermilab (CDF) scientific experiment. Arxiv preprint arXiv, 0810.3453, 2008
3. I. Foster. The grid: A new infrastructure for 21st century science, Physics Today, 55, 42–47, 2002
4. I. Foster and C. Kesselman. Computational grids. Cern European Organization for Nuclear Research-Reports-Cern, 1998, pp. 87–114
5. A. Chervenak, I. Foster, C. Kesselman, C. Salisbury, and S. Tuecke. The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets. Journal of Network and Computer Applications, 23, 187–200, 2000
6. C. Blanchet, R. Mollon, D. Thain, and G. Deléage. Grid deployment of legacy bioinformatics applications with transparent data access. Gene, 153, 224,000
7. M. Turner, D. Budgen, and P. Brereton. Turning software into a service. Computer, 38–44, 2003
8. M.P. Papazoglou. Service-oriented computing: Concepts, characteristics and directions. Proceedings of the Fourth International Conference on Web Information Systems Engineering, Washington: IEEE Computer Society Press, December, 2003
9. R.A. Belderbos and L. Sleuwaegen. Local content requirements and vertical market structure. European Journal of Political Economy, 13, 101–119, 1997
10. R. Ranon, L. De Marco, A. Senerchia, S. Gabrielli, L. Chittaro, R. Pugliese, L. Del Cano, F. Asnicar, and M. Prica. A web-based tool for collaborative access to scientific instruments in cyberinfrastructures. In Grid enabled remote instrumentation. Signals and communication technology, F. Davoli, N. Meyer, R. Pugliese, S. Zappatore, Eds., Springer, New York, NY, pp. 237–251, 2009, http://dx.doi.org/10.1007/978-0-387-09663-6_16.
11. B. Coghlan, E. Fernández, E. Heymann, P. Heinzlreiter, S. Kenny, M. Owsiak, I. C. Plasencia, M. Plóiennik, H. Rosmanith, M. A. Senar, S. Stork, and R. Valles. Int.eu.grid project approach on supporting interactive applications in the grid environment. In Grid enabled remote instrumentation. Signals and communication technology, F. Davoli, N. Meyer, R. Pugliese, S. Zappatore, Eds., Springer, New York, NY, pp. 435–445, 2009, http://dx.doi.org/10.1007/978-0-387-09663-6_28.
12. I. Marco. The interactive european grid: Project objectives and achievements. Computing and Informatics, 27, 161, 2008
13. S. Basu, V. Talwar, B. Agarwalla, and R. Kumar. Interactive grid architecture for application service providers. Proceedings of the International Conference on Web Services (ICWS), 2003
14. E. Laure, S.M. Fisher, A. Frohner, C. Grandi, P. Kunszt, A. Krenek, O. Mulmo, F. Pacini, F. Prelz, and J. White. Programming the grid with gLite. Computational Methods in Science and Technology, 12, 33–45, 2006
15. W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, L. Liming, and S. Tuecke. GridFTP: Protocol extensions to FTP for the grid. Global Grid ForumGFD-RP, 20, 2003
16. L. Skital, R. Slota, D. Nikolov, and J. Kitowski. Grid enabled virtual storage system using service oriented architecture. Proceedings of the 2005 Conference on Software Engineering: Evolution and Emerging Technologies, IOS Press, Amsterdam, pp. 149–159, 2005.
17. D. Thain and M. Livny. Parrot: An Application Environment for Data-Intensive Computing. Scalable Computing: Practice and Experience, 6, 9–18, 2005
18. Wilson, P. (1991). Computer supported cooperative work: An introduction. Kluwer, Dordrecht
19. R. Pugliese, M. Prica, G. Kourousias, A. Del Linz, and A. Curri. Integrating instruments in the grid for on-line and off-line processing in a synchrotron radiation facility. Computational Methods in Science and Technology, 15(1), 21–30, 2009.

20. M. Prica, R. Pugliese, A. D. Linz, and A. Curri. Adapting the instrument element to support a remote instrumentation infrastructure. Remote Instrumentation and Virtual Laboratories, 11–22, 2010.
21. E. Frizziero, M. Gulmini, F. Lelli, G. Maron, A. Oh, S. Orlando, A. Petrucci, S. Squizzato, and S. Traldi. Instrument element: A new grid component that enables the control of remote instrumentation. Proceedings of the Sixth IEEE International Symposium on Cluster Computing and the Grid (CCGRID06)-Volume 00, IEEE Computer Society Washington, DC, 2006