ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA160

ENABLING DATA ANALYTICS AS A SERVICE FOR LARGE SCALE FACILITIES

K. Woods, R. Clegg, N. Cook, R. Millward, Tessella Ltd, Abingdon, UK
F. Barnsley, C. Jones, STFC/RAL, Didcot, UK

Abstract

The Ada Lovelace Centre (ALC) at the Science and Technology Facilities Council (STFC) is an integrated, cross-disciplinary data intensive science centre, for better exploitation of research carried out at large scale UK Facilities including the Diamond Light Source, the ISIS Neutron and Muon Facility, the Central Laser Facility and the Culham Centre for Fusion Energy (CCFE). ALC will provide on-demand, data analysis, interpretation and analytics services to worldwide users of these research facilities.

Using open-source components, ALC and Tessella have together created a software infrastructure to support the delivery of that vision. The infrastructure comprises a Virtual Machine Manager (VMM), for managing pools of virtual machines (VM) across distributed compute clusters; components for automated provisioning of data analytics environments across heterogeneous clouds; a Data Movement System (DMS), to efficiently transfer large datasets; a Kubernetes cluster to manage on demand submission of Spark jobs. In this paper, we discuss the challenges of creating an infrastructure to meet the differing analytics needs of multiple facilities and report the architecture and design of the infrastructure that enables Data Analytics as a Service.

INTRODUCTION

Advanced scientific research facilities in the United Kingdom (UK), such as ISIS, Diamond, Central Laser Facility (CLF) generate huge amounts of data. For example, for the past 4 years Diamond has generated approximately 4PB of data per year. This is expected to rise to 15PB per year within the next two years [1]. Scientists need advanced computing infrastructure, products and services to interpret and manage the data they obtain during their research. STFC's Scientific Computing Department (SCD) manages high performance computing facilities, services and infrastructure to support such research.

However, for many scientists, making use of advanced, high performance computing infrastructure can be difficult. For example, users of such facilities may be expected to know how to select the most appropriate compute resource for their work, how to set up and configure virtual machines, or how to move data between archives and local storage on compute clusters. Without appropriate experience and training, such tasks can be daunting.

ALC, which is part of SCD, has a remit to support STFC's science programme by building capacity in advanced software infrastructure for the handling, analysis, visualisation, integration, modelling, and interpretation of experimental data [2]. In pursuit of that goal, ALC and Tessella have developed new software components, Piezo, the VMM and DMS, designed to hide the complexity of using such infrastructure. Piezo, the VMM and DMS are designed to free researchers and scientists from the complex mechanics of managing datasets and environments, enabling them to focus on the analysis and interpretation of their data.

Piezo has been designed to support researchers using bespoke modelling and simulation codes, often developed by the researchers themselves. The VMM and DMS are software infrastructure components designed to work in together in support of researchers using mainstream data analysis tools.

Piezo, the VMM and DMS have all been created using readily available open-source technologies.

PIEZO

Scientists at CCFE have access to a single Spark [3] cluster for running modelling and simulation codes. Unfortunately, a small number of large codes can easily consume all of the available cluster resources, meaning that smaller codes, especially those in development, are effectively shut out. CCFE scientists needed a way to run experimental jobs, easily, with guaranteed availability and rapid turn-around. STFC has compute resources that it can make available to CCFE scientists; the problem faced by ALC and Tessella was how to make those resources available to CCFE scientists, while shielding them from the mechanics of transferring data and job management on remote resources.

Our solution, called Piezo, creates on-demand, single-user Spark clusters. The process of spinning up a Spark job is relatively straightforward; spinning up a Spark cluster on demand to process that job requires a lot of detailed technical knowledge, which many scientists do not possess. To orchestrate the management of these Spark clusters, we chose Kubernetes [4], because it provides facilities which simplify the automated set up of the Spark cluster. Piezo operates on two underlying systems:

1. a high-performance compute cluster.
   - a Kubernetes instance running the cluster to administer Spark jobs.
2. a storage platform with an S3 interface.

Spark jobs are run as Docker [5] containers to ensure consistency and repeatability between different execution platforms. We created a simple web Application Programming Interface (API) to shield scientists from the arcane technical details of creating and managing individual Spark jobs. The web API is responsible for:

- creating Spark jobs.
- managing Spark jobs.
- managing the job results and output files.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA160

- informing the user of the job state and location of results.
- security.

In principle, any distributed storage system supported by Spark could be used as the storage infrastructure. The Hadoop Distributed File System (HDFS) [6] is a common choice for Spark clusters. Under HDFS data is stored locally to the Spark cluster, providing good performance. However, we chose to store input and output files on STFC's Ceph [7] storage system, using an S3 API. Using an S3 API means data does not have to be stored locally with the Spark cluster - S3 decouples the storage infrastructure from the compute infrastructure. The performance, compared to HDFS, is slower, but with ever increasing data volumes, the superior scalability of S3 makes it the better choice [8].

To manage the modelling and simulation codes, Piezo uses the Harbor [9] container registry. Harbor provides a simple way to create and manage a library of approved, trusted modelling and simulation codes (in the form of Docker containers). Scientists can administer the registry themselves, creating containers for internally developed codes or for approved codes of external origin.

Fig. 1 illustrates the architecture of Piezo:

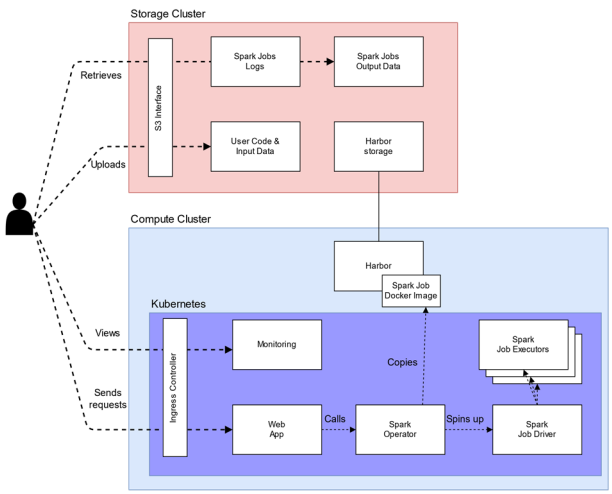

Figure 1: Piezo Architecture.

Piezo provides high-level job monitoring using Prometheus [10]. This information is needed by system admin personnel, to ensure that the compute cluster has sufficient resource to support the demands being placed upon it.

For more detailed monitoring of jobs, we expose the Spark UI to scientists via the Kubernetes Ingress rules to obtain detailed metrics of the performance of Spark. This feature of Piezo allows scientists to optimise their own Spark jobs within their own execution environment.

Piezo is a powerful general mechanism built on Kubernetes. It need not even run Spark, which is merely a convenient means for running jobs in parallel. Other frameworks could be used in its stead, or none at all (leaving Kubernetes to manage the scheduling the execution of Docker containers directly).

VIRTUAL MACHINE MANAGER

The Virtual Machine Manager (VMM) is a utility to manage the scaling of demand for computer resources in a cloud. The VMM is built on OpenStack [11] and libcloud [12] and provides an automated, cross-cloud mechanism for managing pools of virtual machines.

The VMM is designed to manage multiple pools of virtual machines. Each pool contains a configurable number of identical virtual machines. Each virtual machine is pre-provisioned with an analysis environment and the tools researchers require to analyse their experimental data. Different pools contain virtual machines configured for different types of analysis. For example, ALC has created pools of virtual machines to support the execution of specialist codes for analysis of neutron and x-ray scattering data. Other pools can contain other types of virtual machine configured to suit different computational needs (e.g. general-purpose machines, machines with many cores, machines with large amounts of memory, GPUs, etc.). It is a straightforward matter to create new pools of virtual machines configured for other types of analysis. The VMM provides a web API to orchestrate requests to allocate and manage virtual machines.

To perform an analysis, researchers need only select (via a higher-level GUI) the dataset they wish to analyse and the type of virtual machine most appropriate to their needs. The VMM will allocate, from the appropriate pool, a virtual machine of the selected type. The selected dataset is automatically mounted on the virtual machine (see DMS below). This process happens without further intervention from the researcher. The researcher needs no special competence in creating virtual machines or moving data across the network. Because the virtual machines are pre-provisioned they are quick to load and be ready for use.

DATA MOVEMENT SYSTEM

When scientists perform data analysis, the data must first be moved to where it is needed, namely the compute cluster. As data volumes grow, the efficient transfer of data between archives (typically medium-term or long-term storage locations) and compute clusters is becoming a pressing problem. There is a balance to be achieved between the availability and cost of short-term data storage on compute clusters and the latency of moving data from archives. The Data Movement System (DMS) addresses this problem transferring only data that is needed in a timely, efficient manner.

The DMS uses a FUSE [13] client on the virtual machine to provide a Unix-like virtual file system. It provides a file/directory view to the end-user (scientist) but, crucially, does not transfer any files until requested to do so (e.g. by a scientist using an application to open a file). The DMS supports file transfer in both directions - archived data can be transferred to the compute cluster and processed data can be saved back to the archive.

Our initial choice for the core data transfer technology of the DMS was GridFTP [14]. It was easy to integrate and provided acceptable performance. However, GridFTP's

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA160

plug-in mechanism did not support the setting of file permissions at the client end. The ability to preserve file permissions on transfer is a crucial requirement to respect the confidentiality of data belonging to research groups. This led us to create a second implementation of the DMS, featuring a custom data transfer mechanism, written in Python, using parallel, secure web-sockets. We have full control of file permissions and the solution is highly scalable.

The second implementation of the DMS also features data transfer adapters. We have created data transfer adapters to:

- automatically transfer all files smaller than a defined (configurable) size. Thus, many relatively small files can be transferred quickly, in the background, with minimal perceived delay.
- transfer all files when a parent directory is opened. This means that while there might be a short delay while transferring the first file, other files are transferred while the user is still viewing the first.
- transfer all files with a common file name element. This allows related files to be transferred together (for example, all files sharing an experiment ID).

Facilities can choose which adapters to use to best suit their own local circumstances. The data transfer adapter mechanism is open, allowing facilities to create their own adapters to suit local circumstances.

CONCLUSION

In Piezo, we have created a solution to allow scientists to submit modelling and simulation jobs, on-demand to a Spark cluster running under Kubernetes. It provides an environment which permits jobs to be run at any time, on any available hardware in a repeatable, containerised fashion. The mechanics of job submission and management are hidden from the scientists, freeing them to focus on calculations and data analysis.

The VMM and DMS work in tandem as key elements of infrastructure supporting pools of virtual machines which provide data analysis environments and tools. The VMM and DMS are adaptable, configurable components, enabling ALC to provide flexible, responsive data analytics services to users of STFC computing facilities.

ACKNOWLEDGEMENTS

We acknowledge funding from UK Research and Innovation -STFC (UK SBS IT 18160).

REFERENCES

[1] N. Cook, private communication, May 2019

[2] Scientific Computing Department Strategy 2017-2021, https://www.scd.stfc.ac.uk/Pages/SCD-Strategy.aspx

[3] Spark, https://spark.apache.org/

[4] Kubernetes, https://kubernetes.io/

[5] Docker, https://www.docker.com/

[6] Hadoop Distributed File System, https://hadoop.apache.org/

[7] Ceph, https://ceph.io/

[8] P. Mrowczynski, "Scaling cloud-native Apache Spark on Kubernetes for workloads in external storages", Masters Thesis, School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden, 2018

[9] Harbor, https://goharbor.io/

[10] Prometheus, https://prometheus.io/

[11] OpenStack, https://www.openstack.org/

[12] libcloud, https://libcloud.apache.org/

[13] FUSE, https://github.com/libfuse/libfuse

[14] GridFTP, https://en.wikipedia.org/wiki/GridFTP