Towards Open Science with Multi-Cloud Computing using Onedata

Michał Orzechowski[1], Michał Wrzeszcz[1], Bartosz Kryza[1],
Łukasz Dutka[1], Renata G. Słota[1,2], Jacek Kitowski[1,2]

[1] Academic Computer Centre CYFRONET AGH, Nawojki 11, 30-950 Kraków, Poland
[2] AGH University of Science and Technology, Faculty of Computer Science,
Electronics and Telecommunications, Institute of Computer Science,
Al. Mickiewicza 30, 30-059 Kraków, Poland
morzech@agh.edu.pl, wrzeszcz@agh.edu.pl,
bkryza@agh.edu.pl,lukasz.dutka@cyfronet.pl,
rena@agh.edu.pl, kito@agh.edu.pl

Keywords: multi-cloud, cloud computing, data access, distributed systems, open science

1. Introduction

Running and deploying applications on different and varied infrastructures is still a major difficulty, particularly for scientific applications which often take the form of intricate workflows made up of multiple steps and require specialized software or libraries, and the use of a Workflow Management System (WMS). Solutions such as virtualization, application containers, and infrastructure automation can help to alleviate this complexity.

Cloud providers offer advanced tools for setting up infrastructure and deploying applications. However, the variety of cloud solutions, such as OpenStack, AWS, or GCP, can make it hard to avoid vendor lock-in. Kubernetes [1] offers a mature set of well-standardized principles and practices for running software in a distributed virtualized environment, which can help to solve this problem. Recently, Kubernetes has been referred to as a "Cloud Native Operating System" in a number of articles, and major cloud providers and platforms now offer "one-click" Kubernetes deployments.

In this study, we present an ongoing project that aims to automate the deployment and execution of scientific workflows. This kind of applications is getting more and more popularity in the field of computational science, including simulations and big data problems in different domains, like for example metallurgy, high energy physics, biotechnology, medicine and others. Our approach utilizes Kubernetes and Onedata data management systems to eliminate differences between various infrastructure providers. The main contributions of this work include the following:

1) we provide an initial implementation of Kubernetes native support for running scientific workflows on multiple cloud infrastructures,
2) we offer a unified solution that allows for data to be delivered to the scientific workflow transparently,
3) we present an architecture of a solution that automates the entire workflow lifecycle, including provisioning, deployment, execution, and monitoring.

2. Onedata Data Management System

Onedata [2,3] is a globally distributed, high-performance data management system that offers transparent and unified access to storage resources worldwide. It can be used for various purposes,

The publication is co-financed from the state budget under the programme
of the Minister of Education and Science called "Excellent Science" project no. DNK/SP/548041/2022

Doskonała Nauka
Ministry of Education and Science
Republic of Poland
Republic of Poland

such as personal data management and data-intensive scientific computations. The fully distributed architecture of Onedata enables creation of hybrid-cloud infrastructure deployments that include private and commercial cloud resources. With Onedata, users can share, collaborate, publish data and perform high-performance computations on distributed data. Furthermore, it supports POSIX--compliant data access for applications.

Onedata is composed of three primary services: Onezone, Oneprovider, and Oneclient. Onezone manages authorisation and metadata distribution, giving users access to the Onedata ecosystem. Oneprovider serves data to users and connects storage systems to Onedata, while Oneclient allows for seamless POSIX-compatible data access on user nodes. Oneprovider can be deployed as a single node or an HPC cluster and can handle large amounts of data with high speeds, even on parallel storage solutions with petabytes of data and GB/s throughput.

Recently it has been featured with a robust workflow engine, powered by OpenFaas [4], which enables creation of advanced data processing pipelines that can access distributed data seamlessly. The workflow feature can be used to establish a comprehensive data archiving and preservation system in compliance with OAIS standards, including ingestion, validation, curation, storage, and publication. The workflow function library provides pre-built functionalities (implemented as Docker images) for typical archival tasks such as metadata extraction, format conversion, checksum validation, virus checks, and more. Custom functions can be added and shared among user groups easily. For further flexibility, to enable transparent data access from the Onedata ecosystem, the functions that execute legacy code can be accompanied by Oneclient, which exposes data using the abstraction of the POSIX filesystem. In case of new implementations, functions can leverage Oneprovider API to fetch data directly via REST.

3. Multi-Cloud Deployment and FaaS Integration

Onedata makes it simple to establish a multi-cloud architecture with seamless access, as shown in Figure 1. It provides also means to expose data located on not cloud-native, legacy storage systems. The data is indexed by Oneprovider and made it globally accessible through Onezone. From the data perspective, Oneprovider connected to a legacy storage system functions as a source Oneprovider, as it holds the original replica of the data. To effectively process data from a legacy storage system on an external cloud, we deploy another instance of Oneprovider on Cloud A, which is connected to a cloud-native storage system that functions as a cache for replicated data. When accessing the data, missing data blocks are automatically replicated between Oneproviders. Additional Oneproviders can be deployed on clouds where we want to process data, creating a network of caching Oneproviders. The data is replicated between them depending on which Oneprovider stores the necessary replica.

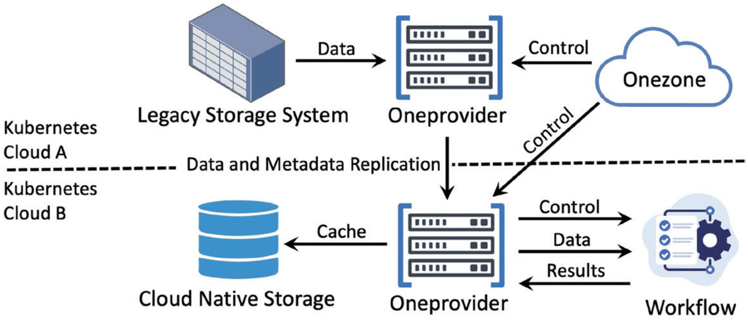

Figure 1. Data Transfer and Workflow Execution on Multi-Cloud.

On a cloud where computation takes place, on a Kubernetes cluster an instance of OpenFaas is deployed next to Oneprovider. The scientific workflow, that uses data stored on Cloud A can be executed on Cloud B. Oneprovider manages the transfer of needed data blocks from Cloud A, caches them on Cloud B and then transfers them to individual OpenFaaS function, used to realize the workflow execution. This architecture highly leverages the mature ecosystem of Kubernetes that enables easy multi-cloud execution of scientific workflows, independently of physical location of data.

4. Conclusion and future work

The proposed multi-cloud architecture and mechanism for executing scientific applications using scientific workflows can greatly simplify the automation of complex workflow scenarios such as:

1) Reproducibility: upon completion, the full execution state (including the runtime environment) of the workflow is recorded and can be used to repeat the execution in the future.
2) Live migration: if the workflow needs to be moved to a different computing infrastructure, its execution state can be used to restore the environment and resume the workflow after migration is complete.
3) Fault tolerance: in case of failure due to infrastructure or software error, the workflow execution can be resumed as soon as the cause of the failure is resolved.
4) Smart-rerun: as the state of each execution step of the workflow is recorded (including intermediate data), parts of the workflow can be replaced and re-run, allowing the workflow engine to reuse already computed data or repeat the execution as needed.

The results show feasibility of the proposed solution and confirm that under certain conditions the proposed new approach for making use of legacy data systems can lead to improvements in workflow execution time. Future work includes automating deployment of Kubernetes clusters on different cloud providers and improvement of existing procedures of deploying Onedata and OpenFaas components, based on supplied requirements and environments characteristics.

References

1. Container Orchestration Kubernetes: https://kubernetes.io.
2. Wrzeszcz M., Kitowski J., Słota R.G.: *Towards Transparent Data Access with Context Awareness*. Computer Science, 19, 2018, 201–221.
3. Wrzeszcz M., Dutka Ł., Słota R.G., Kitowski J.: *New approach to global data access in computational infrastructures*. Future Generation Computer Systems, 125, 2021, 575–589.
4. Cloud Operating system.
5. https://blogs.oracle.com/cloud-infrastructure/kubernetes-a-cloud-and-data-center-operating-system.
6. OpenFaaS – Serverless Functions Made Simple: https://openfaas.com.

Acknowledgements. MO and JK are grateful for support from Proteus-RS project (EU RFCS grant, contract 899455). RGS is grateful for support from the subvention of Polish Ministry of Education and Science assigned to AGH University. This scientific work was supported in part by an international project co-financed by the program of the Minister of Science and Higher Education entitled “PMW” in the years 2020-2021; contract No. 5145/H2020/2020/2.