Automated and portable Hadoop cluster orchestration on clouds with Occopus for Big Data applications*

Enikő Nagy[1,2], József Kovács[1], Róbert Lovas[2]

[1]Institute for Computer Science and Control, Hungarian Academy of Sciences, Budapest
[2]John von Neumann Faculty of Informatics, Óbuda University, Budapest

eniko.nagy@sztaki.mta.hu, jozsef.kovacs@sztaki.mta.hu,
lovas.robert@nik.uni-obuda.hu

Keywords: Cloud computing, Orchestration, Hadoop, Occopus, Contextualization

1. Introduction

Apache Hadoop [1], an open-source software framework for storing data in a distributed cluster environment and running applications to process this large amount of data in a fast and efficient way. In the last few years, Hadoop has become a very popular system for analyzing Big Data with its MapReduce [2] framework introduced by Google in 2004. Many scientific applications, such as weather forecasting [3], DNA sequencing [4], and molecular dynamics [5], have been parallelized using Hadoop. However, the deployment of a fully functional Hadoop cluster is not a trivial task, it is currently not in line with the capabilities of the data scientists, and therefore there is still a significant barrier for this technology to spread among data scientists.

Combining Hadoop, Cloud and an orchestration tool for dynamically build up Hadoop clusters would help these scientists run their Big Data applications. Complex virtual infrastructures, like Hadoop, with all of its configuration and network design, needs special planning, care and skills by the end-users to have proper functioning Hadoop cluster. One of our main targeted user groups is the Hungarian academic research community and their new computing infrastructure, the MTA Cloud.

This paper focuses on utilizing a hybrid, cloud orchestration tool called Occopus [6]. The solution presented in this paper, provides automatic deployment of a fully functional Hadoop cluster without the need for low level understanding of Hadoop architecture or cloud computing. Moreover, (1) it is portable, since the solution does not depend on any cloud-specific feature, (2) it is scalable by utilizing Occopus and Hadoop dynamicity, (3) it does not require any prepared image, (4) it gives the possibility to fine-tune the configuration of the Hadoop components for advanced users and finally (5) it supports short or long-term usage scenarios.

2. Implementation

Occopus is a hybrid, cloud orchestration tool developed by the Institute for Computer Science and Control, Hungarian Academy of Sciences (MTA SZTAKI). It is an open source software providing features for configuring and orchestrating distributed virtual infrastructures both on single and multi-cloud environments. Occopus operates based on descriptors that describes the infrastructure layout, the individual nodes, the resources to be used, the configuration management details, the contextualization of the nodes and the way the services

* This work is supported by the National Research, Development and Innovation Fund of Hungary under grant No. VKSZ_12-1-2013-0024 (Agrodat), and by the International Science & Technology Cooperation Program of China under grant No. 2015DFE12860.

on the nodes can be monitored. Once this information is provided, Occopus is able to build, maintain, scale and destroy the infrastructure. As a summary, Occopus is a lightweight, easily deployable and usable orchestration tool with high level of flexibility and cloud-independence.

In our solution, there are one Hadoop Master and several Hadoop Slave nodes in the Hadoop cluster. All the nodes are deployed automatically with Occopus, based on the descriptors. Occopus can carry out node contextualization based on cloudinit [7], therefore Hadoop nodes are built based on images without preinstalled software to avoid additional dependency on images. Configuration and settings for a properly working Hadoop cluster is deployed by cloudinit. The dynamicity of the Hadoop cluster is based on service discovery provided by the Consul tool. Each Hadoop slave node is registered under Consul to provide cluster information for the Master node about the actual cluster. With Occopus user can perform up and downscaling on-demand, manually. The deployed Consul with some helper module performs the dynamic reconfiguration of the Hadoop cluster. Depending on the performance of the cloud scaling procedures are rather quick as upscaling is performed parallel by Occopus, while downscaling requires few resource. Based on our measurements, the presented solution scales well and the applications scale up together with the performance of the Hadoop cluster dynamically.

3. Conclusion and future work

This paper presented a solution of a fully automatized, scalable Hadoop cluster implementation by Occopus on cloud. Data scientists of MTA Cloud can use this solution to create complex, short or even long time life cycle virtual Hadoop infrastructures for scientific projects. Thanks to the Occopus tool, this solution works on the wide range of popular clouds (with EC2, Nova and other APIs), and does not depends on precompiled images and proprietary management solutions offered by cloud providers as black box services.

As future work, to prevent the possibly occurring data loss, an automatic protection for scaling down too fast will be created. We can scale down Hadoop cluster when there is not enough load to prevent the waste of virtual resources. Due to the fact that each node holds a piece of the data and the default configuration is to have 3 replicas of each block, once the 3 nodes holding those replicas are shutdown the data is lost. The safest way is to turn off instances one by one and after turned off instance notifying the NameNode about the decommissioned node, forcing data replication to other nodes. After that, a balancer tool, provided by Hadoop, can be used to re-balance a changed cluster. Furthermore, in the future we plan to implement an automatic scaling (up) of an overloaded virtual Hadoop infrastructure.

References

[1] Apache Hadoop. http://hadoop.apache.org/

[2] Dean J, Ghemawat S. MapReduce: Simplified Data Processing on Large Clusters. Commun. ACM Jan 2008; 51(1):107–113, doi:10.1145/1327452.1327492.

[3] Li L, Ma Z, Liu L, Fan Y. Hadoop-based ARIMA Algorithm and its Application in Weather Forecast. International Journal of Database Theory and Application 2013; 6(5):119–132, doi:10.14257/ijdta.2013.6.5.11.

[4] Schatz MC. Cloudburst: highly sensitive read mapping with mapreduce. Bioinformatics 2009; 25(11):1363–1369,doi:10.1093/bioinformatics/btp236.

[5] Jiao S, He C, Dou Y, Tang H. Molecular dynamics simulation: Implementation and optimization based on Hadoop. 2012 Eighth International Conference on Natural Computation (ICNC), 2012; 1203–1207, doi:10.1109/ICNC.2012.6234529.

[6] G. Kecskeméti, M. Gergely, Á. Visegrádi, Zs. Németh, J. Kovács, P. Kacsuk: One Click Cloud Orchestrator: Bringing Complex Applications Effortlessly to the Clouds. In: Euro-Par 2014. Lecture Notes in Computer Science (8806). Springer, pp. 38-49.

[7] Cloud-init. https://cloudinit.readthedocs.io/en/latest/