



HAL
open science

GUESS: monitorinG join qUery Execution in Serverless and Serverful spark

An-Truong Tran Phan, Laurent d’Orazio, Thuong-Cang Phan, Le Gruenwald

► **To cite this version:**

An-Truong Tran Phan, Laurent d’Orazio, Thuong-Cang Phan, Le Gruenwald. GUESS: monitorinG join qUery Execution in Serverless and Serverful spark. International Conference on Database Systems for Advanced Applications (DASFAA), Jul 2024, Gifu, Japan. hal-04544429

HAL Id: hal-04544429

<https://hal.science/hal-04544429v1>

Submitted on 12 Apr 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

GUESS: monitorinG join qUery Execution in Serverless and Serverful spark

An-Truong TRAN PHAN¹, Laurent D’ORAZIO², Thuong-Cang PHAN³, and Le GRUENWALD⁴

¹ Vinh Long University of Technology Education, Vietnam

² Univ Rennes, CNRS, IRISA

³ Can Tho University, Vietnam

⁴ School of Computer Science, The University of Oklahoma

Abstract. This paper proposes a monitoring system called GUESS to compare the performance and energy consumption of join query processing on Spark in Serverless and Serverful environments. The system collects metrics on resource utilization, query execution times, and power usage through Prometheus, Grafana, Spark History Server, and Open-Manage Enterprise Power Manager. These metrics are visualized through an intuitive web dashboard to enable easy comparison between Serverless and Serverful Spark workloads. Experimental results using the TPC-H benchmark show that the Serverless environment consumes less energy than the Serverful environment due to on-demand resource allocation. However, the Serverful environment exhibits better query performance, especially for workloads with known resource requirements. GUESS provides insights into optimizing resource efficiency and query performance when deploying Spark analytic workloads.

Keywords: Serverless computing · Serverful computing · Cold-start · Warm-start · Query Execution · Spark

1 INTRODUCTION

The increasing energy consumption of data centers poses significant environmental challenges. According to 2023 reports [5], global data center electricity consumption is estimated to range from 240 - 340 terawatt hours (TWh), equivalent to approximately 1 - 1.3% of total global electricity demand. Moreover, this energy consumption also produces about 330 million tons of CO₂, equivalent to 0.9% of total global greenhouse gas emissions. Energy consumption and greenhouse gas emissions are major issues for data centers, necessitating new research and novel computational models to reduce their environmental impact [6].

There are some research studies comparing the performance and energy consumption of computing models in Serverless and Serverful environments. Chang-peng Zhu et al. [8] conducted a study comparing the performance of Spark on bare metal and Kubernetes environments with several algorithms such as SQL

Join, Kmeans, and Wordcount to evaluate the two environments based on various metrics. Anthony Byrne et al. [3] conducted an experiment to compare the energy consumption of the MicroFaaS Serverless model with that of a virtual machine cluster. They used a custom cluster orchestration software to send requests simulating function invocations to the workers. The experiment utilized the WattsUp Pro power meter to measure the energy consumption of both the MicroFaaS cluster, consisting of multiple single-board computers, and the virtual machine cluster running on a physical server. Abdulaziz Alhindi et al. [1] conducted an experiment to compare the energy consumption between the OpenFaaS Serverless model and Docker containers. They used the Jmeter tool to generate workload sent to the benchmark applications and employed the Powerstat energy consumption measurement tool to gauge the power usage of the virtual machines deploying OpenFaaS and Docker. Previous studies have focused on comparing the performance and energy consumption between models in virtualized environments using software tools that often lack accuracy. There has been no research that can monitor and compare the performance and energy consumption of Join operations in a big data environment on Serverless and Serverful environments. Our research introduces a new approach using specialized hardware tools such as OpenManager Enterprise Power Manager, Prometheus, Grafana, and Spark history server to directly measure, monitor, and evaluate the performance and energy consumption when performing Join operations on a big data environment while executing Spark applications in Serverless and Serverful environments.

In this study, we propose GUESS system to allow users to visualize the performance and energy consumption of Spark applications [2] when running in Serverless [4] or Serverful environments. We experimentally evaluate the performance and energy consumption using the TPC-H¹ database benchmark with 22 join query workloads on big data platforms. GUESS system aims to (1) visualize the performance and energy consumption of Spark applications when deployed in Serverless and Serverful environments; and (2) compare the differences in the initialization, data processing, and resource reclamation phases of Spark applications when running in Serverless versus Serverful environments.

2 SYSTEM OVERVIEW

Figure 1 shows the system architecture, which is composed of four subsystems. Users can interact with the system through a web application. The specific functions of each component are described below.

2.1 Spark Application

We write a Spark application in the Scala programming language to conduct experiments evaluating performance and energy consumption. Specifically, 22 standard query templates from the TPC-H benchmark dataset were used, with

¹ <https://www.tpc.org/tpch/>

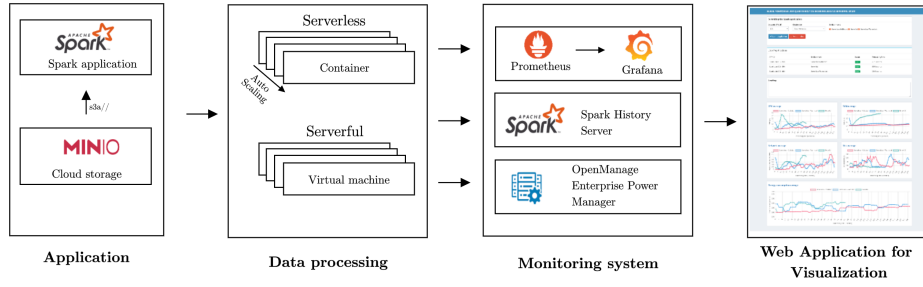


Fig. 1: System overview

dataset sizes ranging from 5GB, 10GB, 25GB, to 50GB. The use of standard queries on large datasets allows for a comprehensive assessment of the Spark application’s processing performance and energy consumption in both Serverless and Serverful environments.

2.2 Data Processing

We run the Spark application in both Serverless and Serverful environments to monitor the differences in cluster initialization, data processing, and resource reclamation stages. Computational resource usage and energy consumption are also recorded for a detailed comparison between the two environments. The Serverful cluster is deployed on virtual Ubuntu machines with 5-20 worker nodes. The Serverless environment uses Kubernetes [7] for container orchestration to automate the deployment and scaling of containerized applications. Leveraging autoscaling capabilities, Kubernetes provides resources including CPU, memory, storage, and networking. By running on Kubernetes, Spark simplifies management and scaling. The Serverless environment experiences cold starts, where Spark initializes the environment for the first run, and warm starts on subsequent runs where Spark images already exist.

2.3 Monitoring System

To accurately monitor performance and energy usage in Serverless and Serverful environments, we develop a system with three key components:

- **Prometheus:** collects system metrics over time while *Grafana* visualizes them through dashboards with charts, graphs, and alerts. This enables real-time monitoring of resource utilization;
- **Spark History Server:** tracks details about executed Spark jobs, tasks, stages, and processes. It provides historical application performance data;
- **OpenManage Enterprise Power Manager:** measures data center energy consumption. We leverage its analytic capabilities to experimentally evaluate and compare Serverless and Serverful energy usage. The proposed monitoring framework integrates real-time and historical data on system performance and power consumption.

2.4 Web Application for Visualization

We develop a web application that allows users to monitor and track system performance metrics like CPU utilization, RAM usage, and network bandwidth in real time. It also enables comparing the energy consumption in the cold-start and warm-start phases of the Serverless environment versus the Serverful environment.

3 DEMONSTRATION

In our demonstration, we use the TPC-H benchmark dataset to evaluate the performance and energy consumption of the Spark application in Serverless and Serverful environments. To visualize the demo result in more detail, we have posted a short demo video and some sample data on GitHub² for access.

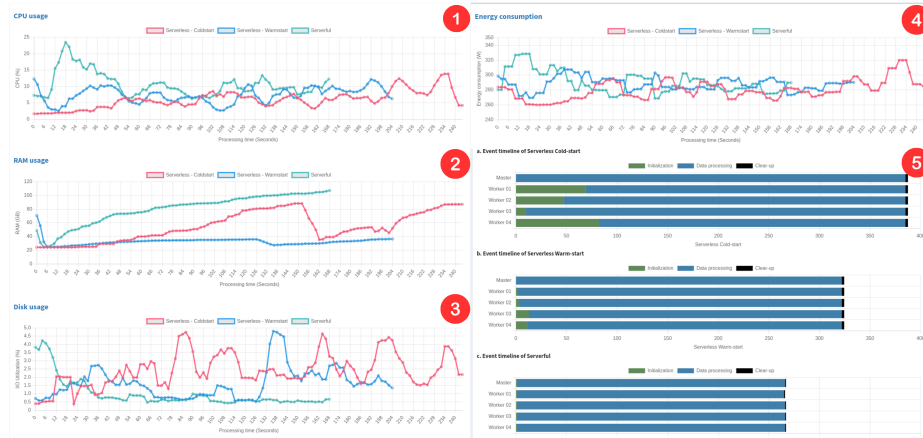


Fig. 2: GUESS monitoring system for visualization

To get started with GUESS system, we first select the TPC-H dataset with sizes ranging from 1-50GB, then choose three execution environments with the Serverless cold-start, Serverless warm-start, and Serverful on the cluster size (5-20 worker nodes). The system will execute the Spark application with 22 sample queries to evaluate performance and display experimental metrics including:

- **Performance:** Figure 2 (1-3) shows resource usage such as CPU, RAM, disk, and network bandwidth. In the Serverless environment, due to the need to initialize the environment for each application execution, network bandwidth and data read/write processes are 2 - 2.3 times higher than in a Serverful environment. Through experiments with extensive workloads, the Serverless environment uses 6.5 - 10.8% less resources compared to the Serverful environment;

² URL of demonstration video: <https://github.com/beta21s/dasfaa2024>

- **Energy consumption:** The Serverless environment, characterized by the need to initialize and reclaim resources after each execution, consumes a certain amount of resources. In contrast, the Serverful environment must consume energy whether or not it is processing data. The experiments in Figure 2 (4) show that the Serverless environment consumes 1.2 - 2.7% less energy than the Serverful environment, which represents a significant amount of energy saving for large data centers;
- **Worker initialization time:** The experimental results in Figure 2 (5) show that the Serverless environment with cold-start takes 75.8 seconds to initialize Spark, while the Serverless with warm-start environment takes 9.5 seconds for each restart. In contrast, the Serverful environment does not waste initialization time thanks to pre-configuration. However, Serverful resources must always be maintained even when the application is not running, leading to time waste. The Serverless environment allows resources to be released immediately after the application stops, enabling them to be used efficiently which leads to a saving in operating costs and energy.

References

1. Alhindi, A., Djemame, K., Heravan, F.B.: On the power consumption of serverless functions: An evaluation of openfaas. In: 2022 IEEE/ACM 15th International Conference on Utility and Cloud Computing (UCC). pp. 366–371 (2022)
2. Armbrust, M., Xin, R.S., Lian, C., Huai, Y., Liu, D., Bradley, J.K., Meng, X., Kaftan, T., Franklin, M.J., Ghodsi, A., Zaharia, M.: Spark SQL: relational data processing in spark. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 - June 4, 2015. pp. 1383–1394. ACM (2015)
3. Byrne, A., Pang, Y., Zou, A., Nadgowda, S., Coskun, A.K.: Microfaas: Energy-efficient serverless on bare-metal single-board computers. In: 2022 Design, Automation & Test in Europe Conference & Exhibition (DATE). pp. 754–759 (2022)
4. Hellerstein, J.M., Faleiro, J., Gonzalez, J.E., Schleier-Smith, J., Sreekanti, V., Tumanov, A., Wu, C.: Serverless computing: One step forward, two steps back (2018)
5. IEA: Data centres and data transmission networks (Sep 2023), <https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks>, Last accessed on 2023-10-25
6. Vahidinia, P., Farahani, B., Aliee, F.S.: Cold start in serverless computing: Current trends and mitigation strategies. In: 2020 International Conference on Omni-layer Intelligent Systems (COINS). pp. 1–7. IEEE (2020)
7. Zhu, C., Han, B., Zhao, Y.: A comparative study of spark on the bare metal and kubernetes. In: 2020 6th International conference on big data and information analytics (BigDIA). pp. 117–124. IEEE (2020)
8. Zhu, C., Han, B., Zhao, Y.: A comparative performance study of spark on kubernetes. *The Journal of Supercomputing* **78**(11), 13298–13322 (Jul 2022)