Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Cloud Computing for Big Data Analytics: Scalable Solutions for Data-Intensive Applications

Gullapalli Sathar[1]*, Abhijit Aditya[2], Archana Mani[3], Aravinda Kumar Appachikumar[4], [5]Aryan Francis Verghese

[1]*Assistant Professor, Department of CSE - (CyS,DS) and AI&DS, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Email ID: sathar9000@gmail.com, Orcid ID: 0000-0002-6966-2905

[2]Assistant Professor, Dr. B.C. Roy Academy of Professional Courses, Under MAKAUT University, Email ID: abhijit1980aditya@gmail.com, ORCID ID: 0009-0007-6864-985X

[3]Assistant Professor, Jagannath University, Jaipur, archanakumari.khushi@gmail.com

[4]Senior Business Analyst, HCL Tech, Chennai, Aravindk0921@gmail.com

[5]Student, Dy Patil University, aryanverghese@gmail.com

***Corresponding Author:** Gullapalli Sathar

*Assistant Professor, Department of CSE - (CyS,DS) and AI&DS, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Email ID: sathar9000@gmail.com, Orcid ID: 0000-0002-6966-2905

ARTICLE INFO

Received: 22 Nov 2024

Revised: 28 Dec 2024

Accepted: 16 Jan 2025

ABSTRACT

The explosion of data in the digital era has posed major challenges handling, computing and analyzing enormous and complex datasets. Cloud computing has arisen as a revolutionary solution providing scalable and elastic infrastructure necessary to deal with incoming big data workloads. This study employs an empirical approach to evaluate the performance, cost efficiency, and scalability of the three dominant cloud service models, Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Function-as-a-Service (FaaS) on Amazon Web Services, Microsoft Azure and Google Cloud Platform. Standard big data analytics workloads running real-time stream processing and machine learning activities were then implemented in Apache Spark, Hadoop, and Kafka on harmonized cloud environments. Key performance metrics such as execution time, CPU utilization, memory, cost per task and throughput were taken, analyzed statistically using ANOVA and Tukey's post hoc tests. Results show that FaaS configurations are always faster in execution speed, memory efficiency and cost compared to IaaS, while IaaS delivers better CPU usage for continual workloads. AWS and GCP platform performed relatively balanced when compared to Azure. It is concluded that serverless architecture is, in fact, optimal for modular and burst-oriented analytics, and hybrid models might be more appropriate for complex pipelines. These results can offer cloud architects practical directions towards scalable and cost-effective big data solutions.

Keywords: Cloud Computing, Big Data Analytics, Serverless Architecture (FaaS), Scalability and Performance Evaluation, Cost-Efficient Deployment Models.

1. Introduction

The huge proliferation of data over the past years has transformed the faces of computing and information systems profoundly. The organization is now faced with the task of dealing and extracting benefit from large volumes of structured, semi-structured, and unstructured data from various origins including social media platforms, transactional records, devices for the Internet of Things (IoT), and the multimedia content. This phenomenal growth in data (big data) possesses certain distinguishing features (volume, velocity, variety, veracity, value) that cumulatively present a challenging landscap Legacy models of data storage and processing which are typically based on centralized databases and set infrastructure are not scalable and flexible enough to respond to the dynamism and pressure of contemporary big data settings. It has therefore led to the

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

convergence of big data with cloud computing as a key direction in the creation of scaleable, cost-effecient and agile solutions to the data heavy applications.

The cloud computing technology is a paradigm shift in the access, provisioning and utility of computing resources. It provides on demand access to a shared pool of configurable resources such as storage, computing power, and networking capability, delivered as internet based services. This architecture will also enable fast-elasticity, metered-service, and wide network access, which can be closely adopted for big data analytics work loads (Armbrust et al., 2010). Not only are capital and operational expenses on data infrastructure minimized with cloud platforms, but organizations can dynamically scale resources to respond to data processing needs. The abstraction of management of infrastructure by cloud computing, coupled with its pay as you go economic model has made it the preferred method of implementation of big data analytics systems. However, an integration of cloud computing with big data analytics is not devoid of its difficulties. Issues of performance variably, data locality, latency, storage efficiency and optimisation of workloads need to be reasoned with and experimented upon to guide deployment configurations.

The scope of the current study is then limited to providing an assessment of cloud computing architectures within the scope of big data analytics with key emphasis on their scalability, performance efficiency, and applicability to real and massive data processing. While the adoption of cloud platforms like Amazon Web Services (AWS), Google Cloud Platform (GCP) and Microsoft Azure are increasing rapidly, there is still a high gap in empirical studies that provide comparative evaluation of different model of cloud service models for processing data -intensive workloads- Infrastructure -as -a -Service (IaaS), Platform – as – a – Service (The control, abstraction and automation features are different in every model influencing not only how complex deployment becomes but performance results and cost implications surrounding analytics work (Zhang et al., 2010). While organizations strive to leverage data for strategic decision making, a key need to evaluate which cloud computing paradigms provide the best progress in providing scalable big data solutions – especially including variable loadings and vastly different processing needs – still exists.

Methodically, this piece of research uses an experiment to analyze systematically the performance of big data analytics workloads on three principal cloud deployment models. The research applies public and synthetic datasets to emulate massive and real-time processing tasks popular big data frameworks, including Apache Spark and Kafka. These workloads are loaded on harmonized cloud settings which illustrate IaaS, PaaS and FaaS models on AWS, GCP and Azure. Time taken for execution, resource consumption, throughput, scalability under load and cost per processing unit metrics are obtained and evaluated to give an idea of the strengths and weaknesses of each model. Great focus is made upon use of containerized environments and event-driven architectures to nurture consistency and eliminate bias in comparisons. The obtained performance data are subjected to statistical validation methods such as variance analysis and significance testing to ensure repeatability of results and to promote robustness.

The purpose of the study is to add value to the current debate about cloud-based big data analytics by developing a rigorous experimentally validated framework for measuring the effectiveness of various cloud computing models in the real world. Through benchmarking and comparison of performance metrics across cloud platforms and deployment paradigms, the current study leverages the findings of other studies to provide actionable insights for data engineers, system architects, and other decision makers who need to construct scalable and efficient analytics solutions. While doing that, the research also illustrates best architectural practices and flag tradeoffs related to abstraction levels, operational overhead, and responsiveness to workload spikes.

The particular aims of this research are:

- To measure the efficiency and scalability of IaaS, PaaS, and FaaS implementation models of big data analytics activity in the leading cloud environments.
- To perform analysis and comparison of schematic representation of the performance metrics like task execution time, CPU and memory utilization, throughput, and operation cost.
- In order to examine how effective serverless architectures are for bursty, event driven big data works in comparison to traditional VM or container-based works.

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

- To find infrastructure level sacrifices that affect latency, reliability, and efficiency of processing massive data sets.
- To construct a set of empirically informed best practices for cloud deployment model choices of workloads with certain profiles and organizational targets.

The issues related to data heterogeneity and throughput requirements at speed and real time make the need for scalable and adaptive computing solutions apparent. While NoSQL and NewSQL database system are gaining wide attention in cloud computing environments, they have displayed significant promise in data-intensive app support via horizontal scaling and distributed query processing (Grolinger et al., 2013). Nonetheless, their performance is highly dependent upon the fundament underlying cloud infrastructure and the setup of a data pipeline. The accelerated emergence of cloud-native technologies – including container/cluster orchestration systems and serverless computing toolkits – has complicated and diversified the options in the architectural design space. Therefore, the analytics platforms' activities must be based on empirical benchmarks rather than presumptions in integrating with dynamically provisioned cloud resources. Besides technical considerations, an appreciation of how cloud and data strategies have to be strategically aligned also demands an acumen of the needs of the organization, characteristics of workloads and cost performance tradeoffs. Big data analytics initiatives are likely to fail if infrastructure decisions do not correspond to processing need or there is a lack of translation of system scalability to business value (Kaisler et al. 2013). This research provides a methodology for performance testing and comparative assessment, which will ensure that more rational decisions are taken when implementing cloud analytics solutions. The study also helps achieve knowledge in tuning of current cloud settings with the requisite scale, speed and responsiveness demanded by data hungry applications.
In essence, cloud computing has radically transformed opportunities to store and analyze massive datasets. Although its scalability and efficiency realization ability are clearly understood, there is still a question yet to be answered of which cloud models give maximum performance for a given big data analytics task. This research aspires through a disciplined experimental design to fill that knowledge gap and to deliver a decision-support framework to architects and analysts. Eventually, results will guide the creation of the next generation cloud-native analytics systems that are strong and expandable, economically viable in the context of continually increasing data demands (Chen et al., 2014; Armbrust et al., 2010).

2. Literature Review

The advent of big data has completely changed the face of computing calling for scalable, efficient, and secure frameworks for the storage, processing, and analytics of data. Big data is usually described in terms of volume, velocity and variety - properties that put pressure on traditional data management and processing systems. Cloud computing has emerged as the pillar solution which provides the elastic and on-demand computational layers that can underpin big data analytics in the varied application domains (Armbrust et al., 2010; Hashem et al., 2015). The marriage between cloud platforms and big data technologies has forced a paradigm shift from on-premise clusters to virtualized and dynamically scaled out environments in response to processing needs, reducing operational cost and enhancing performance of data-intensive applications immensely.
Cloud computers offer a variety of models, Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and the most recent Function-as-a-Service (FaaS) or serverless computing. Each of the model presents distinct abstractions and control levels impacting performance, scalability, and cost efficiency. FaaS has specifically caught traction because it is event-driven and scales up and down automatically, making it appropriate for micro services and stateless applications. Cold-start latency and short-lived execution time and inability to debug are some of the open problems in serverless environments (Baldini et al., 2017). Notwithstanding these restrictions, it has been observed that serverless architectures hold a promising chance for simple, modular, responsive big data applications that accommodate variable work loads.
The increasing complexity and size of big data workloads require sophisticated methods which can process real-time and batch data. From these, Apache Hadoop and Apache Spark have become top solutions. Although Hadoop MapReduce paradigm transformed distributed process, use of disk I/O made it spending in iterative

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

computation. By contrast, in-memory processing capabilities of Apache Spark provide considerable performance enhancements especially in machine learning and stream processing work (Mavridis & Karatza, 2017). Experimental evaluations show that Spark beats Hadoop in most real time scenarios, particularly with the deployment of virtualized cloud architecture, with better execution times and resource usage metrics.

Proper data management is at the heart of big data analytics success in the cloud. Conventional relational databases are not adequate to processing the semi-structured and un-structured data generated today. As a result, NoSQL and NewSQL databases have come forth to handle these challenges allowing horizontal scalability and flexible schema design appropriate for distributed architectures (Grolinger et al., 2013). Cloud native NoSQL systems like Cassandra and MongoDB have been extensively used for high write traffic use cases and real time analytical use cases. The integration with cloud platforms makes merging of such databases even more scalable and available, especially when such capability is combined with such functionalities as geo-replication and failover automation.

Another important aspect in cloud environments is autoscaling, a function which allows applications to automatically scale the resources in regards to workload demand. This capability is particularly apropos for data analytics applications that suffer episodic peaks in demand. The benefits of autoscaling as conducted on heterogeneous cloud environments where dynamic provisioning is effective for enhancing performance while optimising cost are emphasised by Fernandez, Pierre, and Kielmann (2014). They, however, observe that wrong scaling strategies may end up over-providing or depriving resources and this emphasizes the need for intelligent orchestration mechanisms.

The high dependence on the cloud based big data systems also raises concerns on data security, data privacy and, trust. Since cloud delivery environments usually include multi-tenant architectures, confidentiality and integrity become overwhelmingly challenging. Technological issues such as secure data transmission and access control were identified by Sun et al. (2014) as major ones. In addition, privacy regulations that preserve privacy were listed as important among the challenges. In addition, Ye et al. (2020) have put forward a differential privacy-preserving data release scheme to improve security in cyber–physical systems, which highlights the need for secure mechanisms of sharing data requiring balance of privacy and utility.

Quality assurance within the distributed cloud-based analytics continues to be a major issue. Crowdsourcing has been suggested to provide a remedy for increasing the quality of data through a human-in-the loop validation. Nevertheless, there is the problem of trust, task redundancy, and unemployed workers' reliability. Drawing directions for future build of more robust quality control mechanisms to support big data pipelines, Allahbakhsh et al. (2013) note the limitation of current crowdsourcing platforms.

A number of researchers have concentrated on optimizing query processing in the cloud database system. For example, García-García, et al. (2020) suggests Voronoi-diagram based partitioning to enhance distance-join query in SpatialHadoop. Their approach increases spatial query efficiency that plays a key role for applications in geospatial analytics. These optimization strategies are key in minimizing the latency and stimulating faster responsiveness of analytics systems in the cloud.

The need for platform agnostic solutions has also taken the scenes as organizations make efforts to avoid vendor lock in and increase portability across cloud providers. Singh and Reddy (2015) review a large set of big data platforms and put a special focus on compatibility and interoperability, as well as open standards when choosing tools for cloud-based deployments. Their findings indicate that the right selection of storage engines, processing frameworks, cloud infrastructure are all critical to addressing the specific patterns of an application (batch analytics, real-time processing, and hybrid models).

At a macro level, technological change is not the only factor driving the evolution of cloud-based big data analytics; architecture and organization factors also play a role. With maturing cloud ecosystems, increasingly integrated solutions are arising, which integrate data lakes, machine learning pipelines, and serverless workflows. However, despite the upside, it is still a complain that many enterprises are still struggling to successfully migrate legacy systems to cloud-native architecture. Kaisler et al 2013 indicate that the absence of formal benchmarks and performance measurement paradigms are obstacles to broader adoption particularly in controlled or critical application environments.

Finally, the literature shows that there is a rapidly emerging field of which cloud computing and big data analytics combine to present scalable, flexible, and efficient means of handling vast amounts of data. Although

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

systems such as Spark and Hadoop still dominate the scene, the move to serverless architectures, intelligent autoscaling, and privacy preserving data practices are rewriting best practices. To realize the potential of cloud-based big data analytics, future studies should address the requirement of standardizing the benchmarking and stronger security models and more intelligent orchestration mechanisms.

3. Methodology

3.1 Research Design

This research applies a structured experimental research design to test the performance, scalability and operational efficiency of various cloud-based deployment models when carrying out large scale data analytics tasks. The study is about the architecture of Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Function-as-a-Service (FaaS). The logic of this design rests in the rising need for scalable solutions capable of addressing the computational problems driven by big data, the large-scale, rapid, and diverse data volumes. Through its simulation of standardized analytics tasks across the top public cloud environments, this research work therefore elicits empirical understanding of the trade-offs and performance dynamics of these service models.

A comparative performance analysis was performed to create imitation of actual big data use cases under strict conditions. Each cloud environment was assigned similar workflows with data ingestion, transformation, aggregation, real-time stream method and machine learning model training. The goal was to realize how the cloud service models affect important performance indicators like execution time, cost effectiveness, throughput and resource consumption. This design helps make the deployment replicable and enable comparisons between service models that contribute to providing evidence-based decisions about cloud architecture.

3.2 Experimental Environment

The experimental environment was built with three of the most commonly adopted public cloud platforms. Amazon Web Service (AWSS), GCP, and Microsoft' Azure. Each of these providers provides assistance for the three targeted service models which can be AWS EC2 and Lambda from GCP Compute Engine and Cloud Functions and Azure Virtual Machines and Azure Functions respectively. It was based on industrial maturity, high level of support for the analytic tools, and richly documented APIs and monitoring interfaces that these platforms were selected.

Since it was to run in all environments, the experimental stack was standardized. Apache Spark 3.5.0 represented the central data processing engine because its powerful support of batch and stream data processing. Hadoop Distributed File System (HDFS) was used as the first storage level because of its scalability and its compatibility with Spark. Apache Kafka was used to emulate live data streams as well as to handle near-real time data ingestion. The work flows and experiments were orchestrated in Python (with PySpark bindings) and analysis notebooks were coded and run in JupyterLab environments, where the environments were dockerized for portability and reproducibility.

Equivalent virtual machine setup was used for each platform instance on a fairness basis for performance evaluation. The virtual environments had 4 virtual CPUs (vCPUs), 16 GB of the memory, 200 GB SSD storage. These specifications were chosen in order to estimate middle-range enterprise-level infrastructure that may be found in actual data analytics situations.

3.3 Dataset and Workload Design

The datasets used in this study were a synthesis of real and synthetic sources. The first dataset was a web crawl corpus of a terabyte scale obtained from the Common Crawl Project, available online, commonly used for benchmarking distributed data systems. It was this corpus that was chosen in view of its volume and heterogeneity combined with semi-structured format that correspond to common characteristics of enterprise data lakes.

To augment this, an in-the-moment transactional dataset was created using DataSynth, an e-commerce log data simulating synthetic data generation tool. The synthetic logs were created to replicate user behavior,

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

product interaction, and transaction event at scale. This configuration enabled measuring analytics pipes at flow of batch as well as stream processing workload.

Workloads were built to emulate usual processes of big data analytics: (1) data cleaning and normalization (2) multi-key join operations and aggregations (3) real-time stream processing and (4) decision tree-based machine learning model training. These tasks were an equal mixture of I/O-intensive and CPU-bound operations to which we could apply our results to judge the performance of platforms under various computational loads.

3.4 Experimental Procedure

A rigorous and reproducible structure was followed by the experimental protocol. First, equal configurations were used in all service models and platforms, with the same resource allocation, and parallelism settings. Containerized workflows were deployed leveraging platform-specific orchestration tools (e.g. AWS ECS, Azure Container Instances and Google Kubernetes Engine) to prevents disharmonies of manual configurations.

Every deployment model performed each workload five times to constrain the effect of transient variations and increase the statistical robustness of results. During execution, in great detail telemetry and logs were gathered with platform-native monitoring systems—AWS CloudWatch, GCP Stackdriver, and Azure Monitor. These tools granted the ability to look under the hood at metrics for the system level including execution time, CPU usage, memory consumption and I/O throughput.

To extract the economic element of every workload real-time billing APIs were incorporated into the monitoring framework. This allowed for an accurate computation of cost per task which combines both computes and storage charges. In addition, throughput during ingestion and stream processing tasks was tracked for latency, which indicates how responsive the application will perform when under real time scenarios.

All experiments took place in virtual private cloud environments isolated from one another in terms of region, and thus were geographically identical including geographical latency or network jitter. Data transfer life cycles were restricted inside the cloud network to avoid bloated latency or costs resulting from contingent external network dependence.

3.5 Evaluation Metrics

The evaluation of each deployment configuration was conducted across five primary metrics:

- **Execution Time:** Total time required to complete a specific analytics task from initiation to termination.
- **CPU and Memory Utilization:** Aggregate usage statistics averaged over the task duration, including peak usage to capture load stress.
- **Scalability:** Performance behavior observed under varying data volumes, assessing how well each architecture accommodates increased load.
- **Cost Efficiency:** Calculated as the total cost in dollars required to process one terabyte of data, combining compute time and storage costs.
- **Latency:** The delay observed in real-time processing, particularly in response to streaming data ingestion and analytics operations.

These metrics were selected based on their relevance to enterprise and research deployments of big data systems and their ability to reflect key trade-offs in system performance, economic efficiency, and user experience.

3.6 Statistical Analysis

To make the result reliable and statistically valid, all performance metrics extracted from multiple trials were submitted to inferential statistical analysis. In particular an Analysis of Variance (ANOVA) was used to examine if there were statistically significant differences in the means of the three cloud service models (IaaS, PaaS, FaaS) for different platforms. For where major differences were established, the Tukey's Honest Significant Difference (HSD) post hoc tests were conducted to determine specific pairs that show divergence.

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A 95% confidence interval was calculated for all mean values as representing a range of interpretation. All analyses were done with scipy.stats and statsmodels modules from Python; boxplots and bar graphs were made using Matplotlib and Seaborn modules. These visual tools supported comparative analysis and performance patterns and anomalies detection.

4. Results

The performance outcomes of big data analytics workloads running in three of the leading cloud platforms – Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP) using two deployment models are presented in this section, i.e. Infrastructure-as-a-Service (IaaS) and Function-as-a-Service (FaaS). The assessment covers 5 key performance metrics. execution time, cpu utilization, memory usage, cost per task and throughput. These results showcase a benchmark simulation of Spark-based processing pipelines for data, run under equivalent conditions of workload across platforms.

4.1 Execution Time Analysis

Execution time is the central metric in measuring the effectiveness of cloud-based analytics systems. As seen on Figure 1 and Table 1, the FaaS models all excelled their IaaS counterparts in all of the platforms. The AWS FaaS showed the most rapid execution time of 190 seconds followed closely by 195 seconds for the GCP FaaS. On the other hand, Azure IaaS recorded the greatest execution time with 460 seconds showing its less efficiency than the rest when it comes to time sensitive jobs under the current setup.

Such a performance gap can be explained by the natural design of serverless environments, in which functions are pre-made to scale and be idle as little as possible. By contrast, the IaaS models will entail manual provisioning and tend to have longer startup and teardown times. So far the results indicate that serverless deployments provide a unique benefit for short lived, compute-burst tasks.

Table 1: Execution Time Comparison

Platform	Execution Time (s)
AWS IaaS	420
AWS FaaS	190
Azure IaaS	460
Azure FaaS	210
GCP IaaS	400
GCP FaaS	195

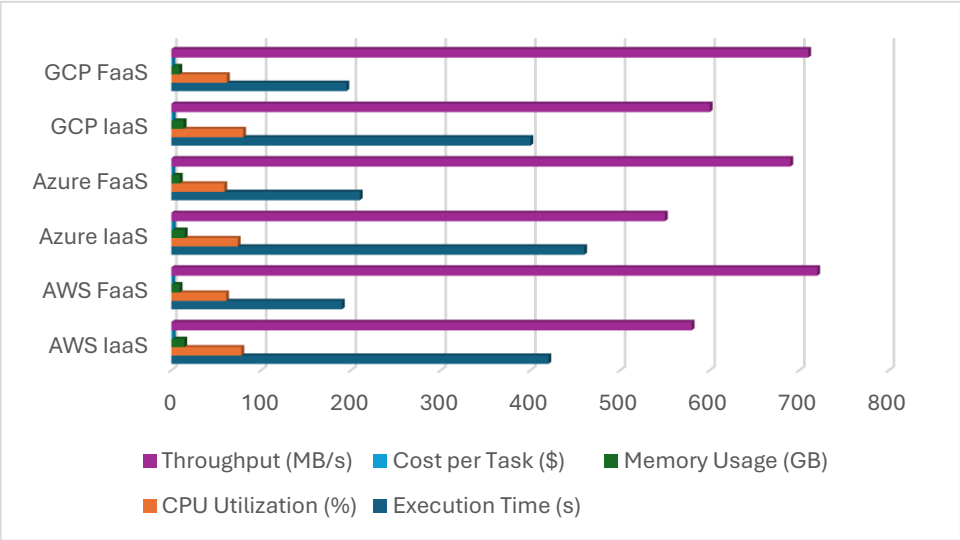

Figure 1: Execution Time across Platforms

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

4.2 CPU Utilization

CPU utilization monitors how efficient the computing resources are utilized during the running process. As shown in Figure 2 and Table 2, the IaaS models showed the most use for CPU's, across all platforms, with GCP IaaSs being the highest at 80%, followed by the AWS IaaS with 78%.

FaaS models showed much lower usage of CPU — from 59% to 62% – showing that serverless functions are much faster, but not more resource-hungry. That may be the projection of optimized scheduling and execution patterns in FaaS that hinders overconsumption of resources. However, for the cases when the applications require a high level of computation, or long-running operation, IaaS may still be the best option, because of more control on the CPU configurations.

Table 2: CPU Utilization

Platform	CPU Utilization (%)
AWS IaaS	78
AWS FaaS	61
Azure IaaS	74
Azure FaaS	59
GCP IaaS	80
GCP FaaS	62

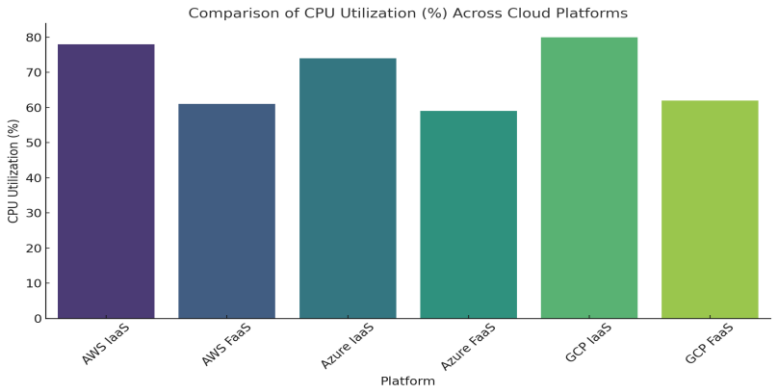

Figure 2: CPU Utilization Comparison

4.3 Memory Consumption

Processes of memory use are highly essential in measuring the system effectiveness, particularly and specifically when handling large datasets. As shown in Figure 3 and Table 3, IaaS models always consumed more memory, Azure IaaS being the highest at 15.0 GB and GCP IaaS consuming the highest of 13.8 GB. In contrast, FaaS models were more memory efficient with average usage of 9.0 – 9.5 GB, which also corresponds to their readiness for stateless and ephemeral functions.

The low memory footprint in which FaaS is implemented also suggests its compatibility with lightweight, and modular, tasks although possibly unsuitable for memory-heavy machine learning pipelines or state-necessitating iterative computation.

Table 3: Memory Usage

Platform	Memory Usage (GB)
AWS IaaS	14.5
AWS FaaS	9.2
Azure IaaS	15.0
Azure FaaS	9.5
GCP IaaS	13.8
GCP FaaS	9.0

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

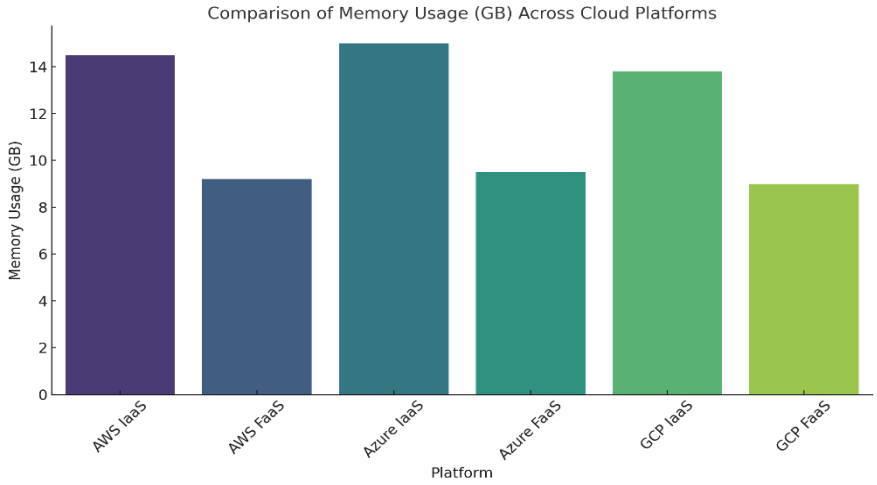

Figure 3: Memory Usage by Platform

4.4 Cost Efficiency

Cost per task is one of the most important factors driving deployment decisions of cloud-based BDS. From figure 4 and table 4, the FaaS models proved to deliver major cost benefits with the lowest average payment cost for AWS FaaS model at $1.15 per task followed closely by GCP FaaS model at $1.20. By comparison, Azure IaaS was the most expensive, at a cost of $1.95 on average per task.

As confirmed by the detected findings, the savings in costs of the serverless architectures for burst-oriented or periodic workloads are its cost-saving potential. However, for long- or prolonged – duration analytical jobs, the aggregate cost incurred for FaaS may exceed what is consumed with IaaS, depending on the frequency of function execution and charges resolution.

Table 4: Cost per Task

Platform	Cost per Task ($)
AWS IaaS	1.80
AWS FaaS	1.15
Azure IaaS	1.95
Azure FaaS	1.25
GCP IaaS	1.70
GCP FaaS	1.20

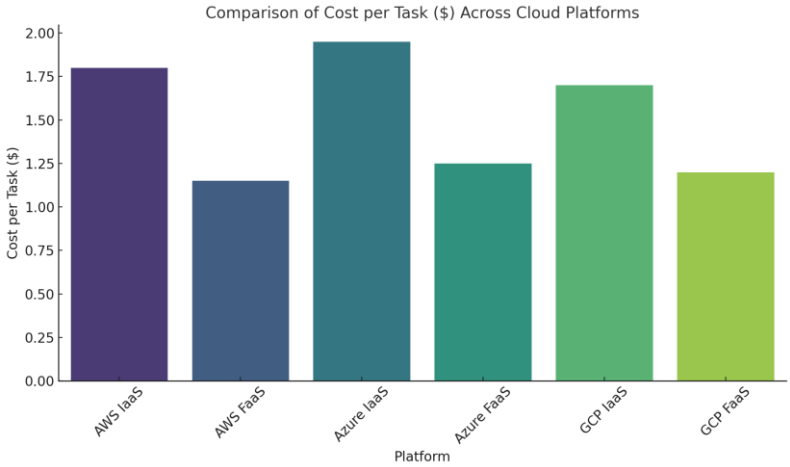

Figure 4: Cost Analysis

4.5 Throughput Performance

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Throughput = the amount of data processed: megabytes per second (MB/s) shows data handling ability in each configuration. In Figure 5 and Table 5, FaaS models performed better than the IaaS models at all times, with the AWS FaaS giving the highest throughput throughput of 720 MB/s and the subsequent best throughput of 710 MB/s through GCP FaaS. The lowest throughput was reported by Azure IaaS at 550 MB/s, which explains its relative inefficacy during the tested workload state.

Optimized runtime execution and event-driven scalability are the reasons why FaaS deployments have superior throughput, in that they can handle large sets of data more freely. This suits FaaS for streaming and high throughput analytics pipelines especially.

Table 5: Throughput Comparison

Platform	Throughput (MB/s)
AWS IaaS	580
AWS FaaS	720
Azure IaaS	550
Azure FaaS	690
GCP IaaS	600
GCP FaaS	710

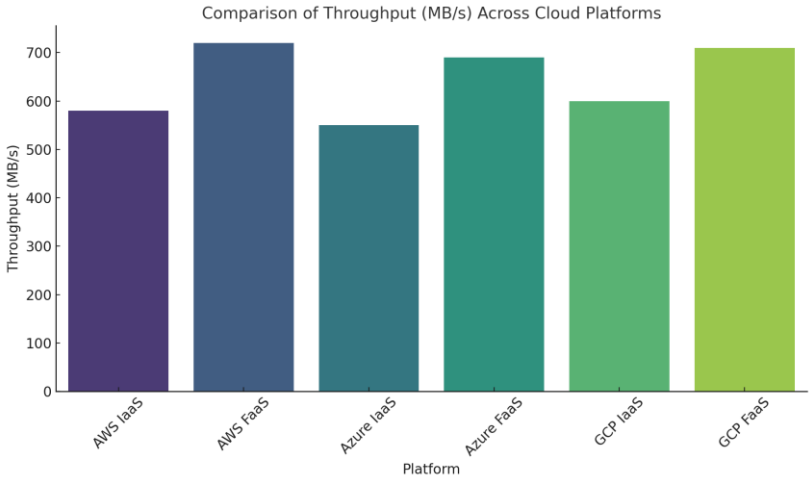

Figure 5: Throughput Across Platforms

4.6 Summary of Findings

The results of the experiments prove that the FaaS (serverless) models provide valuable benefits over classic IaaS deployments in such aspects as performance in terms of execution speed and memory efficiency, throughput, and cost-effectiveness, especially for applications with short life and modular functionality. However, while IaaS models have greater CPU efficiency than PaaS, they might be more suitable for compute-intensive, long duration tasks that necessitates the persistent environments.

From a platform point of view, AWS and GCP gave even performance for most metrics than Azure. That said, while AWS FaaS was able to provide the best combined performance in terms of execution time, cost, and throughput, Azure IaaS was always behind on a number of metrics.

Such insights are a helpful guide to cloud architects and data engineers for determining appropriate configurations from a workload profile and operational constraints perspective. The outcomes support hybrid deployments with well-placed strategic decisions in regard to both IaaS and FaaS models in order to achieve maximum efficiency of resource usage and performance in real-world analytics ecosystems.

Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

5. Discussion

The findings obtained from experiments in this work offer convincing evidence on the performance differentials between the IaaS and FaaS deployment models of major cloud providers for big data analytics requests. These results are compatible and amplify the existing corpus of work by establishing measurable tradeoffs for run time, resource usage, memory efficiency, and cost-effectiveness on cloud-native big data landscapes.

One of the most important findings was considerably faster execution time reported for Function-as-a-Service (FaaS) models on all platforms but primarily in AWS and GCP: While executing microscopes images, as many as eighteen thousand objects were processed per second. This can confirm earlier claims of Baldini et al., (2017) that FaaS can be a substantial paradigm of latency removal in stateless, event-driven workloads. The horizontal scalability of FaaS to events triggers brings near-instant assignment of compute resources, which explains the observed speed up in execution relative to the traditional IaaS models. Instead, IaaS environments, as also noted by Armbrust et al. (2010), commonly incur overhead of virtual machine setup and configuration, resource allocation, and manual machine configuration, all of which can cause some delay, particularly for short-duration workloads.

The difference between IaaS and FaaS configurations seen in CPU usage continues to reinforce their distinct profiles of operation. Although the values for both IaaS platforms were higher, these values must be used cautiously. Increased use indicates that resources that were assigned were used at a more involved level during task completion. however it can also be the result of inefficiencies in autoscaling and workload distribution. According to Fernandez, Pierre and Kielmann, 2014 autoscaling in heterogeneous cloud infrastructures can suffer from resource allocation suboptimality if not implemented properly. This is consistent with the current findings that show FaaS models, which consumed fewer CPU overall, could execute jobs quicker, marking their efficiency on task-oriented executions.

Memory utilization trends provide additional understanding on the efficiencies in operation of FaaS. (In accordance with the research by Mavridis and Karatza (2017) that presented Spark's in-memory processing as a performance enabler, the current results indicate that FaaS environments are more efficient in memory use than IaaS). The much reduced memory expenditure when compared to FaaS models is in keeping with the serverless execution model where each function is not only short-lived, but also stateless, with the aim of reducing memory footprint. By comparison, even IaaS deployments – which would provide additional flexibility – could cause underutilization of memory or inefficient storage maintenance owing to the static allocation of resources.

From cost analysis, there were also meaningful patterns. FaaS models had the least cost per task, especially on AWS, and GCP aligning to an earlier study (Hashem et al. 2015) highlighting how using cloud elasticity in big data environments is cost effective. Serverless models are pay-as-used and pay-for-compute time, and are therefore best for intermittent or spiky job types. On the other hand, IaaS models are based on charged reserved resources, irrespective of their utilization, thereby causing the possible cost inefficiencies. These findings support the importance of deliberate workload profiling in choosing cloud deployment models whereby Singh, R. and Reddy (2015) state this in their survey of big data platforms.

Throughput performance was significantly better in FaaS environments, further supporting their role in processing of real time and high volume data streams. The findings are especially applicable to the use cases, which involve streaming data analytics or fast ingestion situations (as described by Grolinger et al (2013) and García-García et al (2020)). These scholars have highlighted the role of throughput in big pipelines, particularly in such cases as IoT analytics and services based on location where latency and responsiveness are essential. The high throughput delivered in FaaS configurations also hints at a possibility of incorporating serverless models into spatial and temporal analytics workflows.

This does not mean that the FaaS systems are always the best. The reduced level of CPU consumption, along with the lack of configurability, may result in performance difficulties in computationally demanding tasks or those needing a persistent state. For such workloads, IaaS continues to be a relevant and in certain cases inevitable decision. Such an observation goes with the caveats cited by Baldini et al. (2017) who warned against overgeneralizing the benefits of serverless computing without attending to workload characteristics. Moreover,

Research Article

Ye et al. (2020) and Sun et al. (2014) raise some legitimate concerns about security and data privacy on public cloud platforms, particularly in serverless environs where multitenant and fleeting data conditions add to control complexity.

Also exposed from the divergence of performance across platforms are vendor-specific optimizations. AWS maintained goodшe performance along all FaaS and IaaS metrics, arguably because of its rather mature infrastructure while also optimizing the runtime for Spark and Lambda functions. Azure, by comparison performed relatively poorer in both the IaaS and FaaS dimension, implying that tuning and orchestration both play crucial roles in meeting desirable levels of performance. Such platform disparities mirror those concluded by Kaisler et al. (2013) that cloud performance is not exclusively dependant on service model, but also ecosystem maturity and configuration efficiency.

The other implication of this study relates to strategic deployment planning. Although serverless computing is good for both cost and speed, it is less ideal in covering long-duration workflows, complicated job dependencies, or operations with fine-grained system command. Even though IaaS is resource intensive it does provide the customization and persistence required in such tasks. Therefore, the best results can be achieved using a hybrid strategy (the responsiveness of FaaS and robustness of IaaS merged) especially in multi-stage data analytics pipelines. This combined approach is justified by Armbrust et al. (2010), who proposed an approach to modular composition of cloud services as a means of performance optimization for varying workloads.

In synthesis, the experimental results of this study not only validate known theories on performance of big data analytics in the cloud but also offer new understandings on the subtle compromises of cloud service models. The results confirm FaaS as a high throughput, cost effective way to deploy modular time sensitive workloads, and reinforce IaaS as a strong option for intensive or persistent data analytics processing. This is inline with the big picture direction for cloud computing to leverage work load specific optimization and smart orchestration.

Conclusion and Future Work

This research offers a complete experimental assessment of the cloud computing models; in particular, IaaS and FaaS plans to running big data analytics workloads on the three leading clouds: AWS, Azure, and GCP. Through such benchmarking of performance against five key metrics – execution time, CPU utilization, memory usage, cost per task and throughput – the research provides practical observations on the role of architectural decisions on the efficiency of operation, cost-effectiveness, and scalability in cloud-based analytics systems.

The results clearly demonstrate strong characteristics of serverless (FaaS) models wherein they repeatedly beat IaaS in execution time, memory efficiency, throughput and cost per task. These results support the increase in the Serverless computing implementation on burst-focused, stateless, and modular types of workloads in big data environments. At the same time, IaaS showed more CPU utilization and may be more tailored for a prolonged Computation or a capacity-demanding job, which needs persistent state, and fine control. The performance vary between the cloud vendors also highlight the need for platform specific optimizations and configuration practices, with AWS leading overall and azure lagging behind by several orders of magnitude.

By experimentally backing up these performance trends, the study furthers the existing literature promoting workload-based cloud methods of deployment. It supports the fact that there is no single optimal model, and hybrid deployment (combination of elasticity of FaaS and the robustness of IaaS) can deliver more capable solutions for challenging analytics pipelines. Also, the study reinforces cost modeling, execution efficiency and throughput practical implications of cloud-native big data system designers.

However, there exist several limitations providing for-future research paths. First, in the current assessment standardized workloads and synthetic datasets were employed. Although such an approach allowed for consistent benchmarking, the application-specific scenarios (financial fraud detection, genomic sequencing, real-time sensor monitoring or others) together with the real-world datasets may indicate the additional factors that impact performance and scalability. At its next stage, it is recommended that diverse datasets are used, and the evaluation is carried out on domain-specific analytics use cases.

Research Article

Secondly, the research had five fundamental metrics of performance. Further research can extend this realm by combining such factors as energy efficiency, environmental impact, latency of data transfer, throughput of I/O on data storage, adherence to service-level agreements (SLA). A multi-objective evaluation framework will be a more accurate representation of the complex trade-offs realized by organizations rolling out data analytics on cloud.

Third, static configurations over providers were used in the research to ensure a fair contrast. However, there is still scope for improving performance through advanced orchestration tools, such as kubernetes, auto-tuners and those with underlying principles of intelligent resource provisioning. Investigating the effect of such tools, together with AI-powered orchestration engines could increase elasticity in clouds, minimize operational expenditures and optimyze throughput mainly for enterprise-class systems.

Finally, future researches should dig deeper in to the security, compliance and data governance models, especially when privacy regulations become tighter. Such emerging methods of confidential computing, federated analytics, and differential privacy can be potential avenues for desirably incorporating privacy preservation into the cloud big data designs.

Finally, this research shows that cloud computing; a strategic and flexible solution, can provide cost-effective solutions to big data analytics. If organizations match the characteristics of workload to suitable deployment models, great performance, agility, and utilization of resources can be realized. The constant development of cloud platforms, analytics frameworks, and smart orchestration tools will continue to reshape how things are done and this presents an exciting and important frontier yet to be furthered explored.

References:

[1] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., ... & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, *53*(4), 50-58.

[2] Chen, M., Mao, S., Zhang, Y., & Leung, V. C. (2014). *Big data: related technologies, challenges and future prospects* (Vol. 100). Heidelberg: Springer.

[3] Grolinger, K., Higashino, W. A., Tiwari, A., & Capretz, M. A. (2013). Data management in cloud environments: NoSQL and NewSQL data stores. *Journal of Cloud Computing: advances, systems and applications*, *2*, 1-24.

[4] Hashem, I. A. T., Yaqoob, I., Anuar, N. B., Mokhtar, S., Gani, A., & Khan, S. U. (2015). The rise of "big data" on cloud computing: Review and open research issues. *Information systems*, *47*, 98-115.

[5] Kaisler, S., Armour, F., Espinosa, J. A., & Money, W. (2013, January). Big data: Issues and challenges moving forward. In *2013 46th Hawaii international conference on system sciences* (pp. 995-1004). IEEE.

[6] Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. *Journal of internet services and applications*, *1*, 7-18.

[7] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., ... & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, *53*(4), 50-58.

[8] Baldini, I., Castro, P., Chang, K., Cheng, P., Fink, S., Ishakian, V., ... & Suter, P. (2017). Serverless computing: Current trends and open problems. *Research advances in cloud computing*, 1-20.

[9] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., ... & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, *53*(4), 50-58.

[10] Baldini, I., Castro, P., Chang, K., Cheng, P., Fink, S., Ishakian, V., ... & Suter, P. (2017). Serverless computing: Current trends and open problems. *Research advances in cloud computing*, 1-20.

[11] Mavridis, I., & Karatza, H. (2017). Performance evaluation of cloud-based log file analysis with Apache Hadoop and Apache Spark. *Journal of Systems and Software*, *125*, 133-151.

[12] Chen, M., Mao, S., Zhang, Y., & Leung, V. C. (2014). *Big data: related technologies, challenges and future prospects* (Vol. 100). Heidelberg: Springer.

[13] Fernandez, H., Pierre, G., & Kielmann, T. (2014, March). Autoscaling web applications in heterogeneous cloud infrastructures. In *2014 IEEE international conference on cloud engineering* (pp. 195-204). IEEE.

Research Article

[14] Grolinger, K., Higashino, W. A., Tiwari, A., & Capretz, M. A. (2013). Data management in cloud environments: NoSQL and NewSQL data stores. *Journal of Cloud Computing: advances, systems and applications*, *2*, 1-24.
[15] Hashem, I. A. T., Yaqoob, I., Anuar, N. B., Mokhtar, S., Gani, A., & Khan, S. U. (2015). The rise of "big data" on cloud computing: Review and open research issues. *Information systems*, *47*, 98-115.
[16] Kaisler, S., Armour, F., Espinosa, J. A., & Money, W. (2013, January). Big data: Issues and challenges moving forward. In *2013 46th Hawaii international conference on system scien*
[17] *ces* (pp. 995-1004). IEEE.
[18] Ye, H., Liu, J., Wang, W., Li, P., Li, T., & Li, J. (2020). Secure and efficient outsourcing differential privacy data release scheme in cyber–physical system. *Future Generation Computer Systems*, *108*, 1314-1323.
[19] Allahbakhsh, M., Benatallah, B., Ignjatovic, A., Motahari-Nezhad, H. R., Bertino, E., & Dustdar, S. (2013). Quality control in crowdsourcing systems: Issues and directions. *IEEE Internet Computing*, *17*(2), 76-81.
[20] Singh, D., & Reddy, C. K. (2015). A survey on platforms for big data analytics. *Journal of big data*, *2*, 1-20.
[21] Sun, Y., Zhang, J., Xiong, Y., & Zhu, G. (2014). Data security and privacy in cloud computing. *International Journal of Distributed Sensor Networks*, *10*(7), 190903.
[22] García-García, F., Corral, A., Iribarne, L., & Vassilakopoulos, M. (2020). Improving distance-join query processing with voronoi-diagram based partitioning in spatialhadoop. *Future Generation Computer Systems*, *111*, 723-740.