Magna Scientia Advanced Research and Reviews
eISSN: 2582-9394
Cross Ref DOI: 10.30574/msarr
Journal homepage: https://msarr.magnascientiapub.com/

(RESEARCH ARTICLE)

Check for updates

CLOUD-BASED DATA ARCHITECTURE DESIGN FOR SCALABLE ENTERPRISE INFORMATION SYSTEMS

Raghu Praneeth Akula * and Manas Kumar Mohanty

Independent Researcher, USA.

ORCID Details
Raghu Praneeth Akula: 0009-0007-1306-1477

Magna Scientia Advanced Research and Review, 2026, 10(02), 297–303

Publication history: Received on 30 January 2024; revised on 21 March 2024; accepted on 28 March 2024

Article DOI: https://doi.org/10.30574/msarr.2024.10.2.0045

ABSTRACT

As organizations continue to expand their data volumes, and the need for real-time decision-making rises, the structural restrictions of monolithic and on-premises information systems have become increasingly apparent. This paper introduces a layered, cloud-based data architecture that could be used to enable elastic scalability, high availability and data governance across a large enterprise environment. The design stack consists of an API gateway tier, microservices running in containers, event-driven data integration tier and polyglot distributed storage tier, all of which are deployed on elastic cloud infrastructure. The evaluation is performed using a simulation based approach which is compared to a traditional monolithic deployment along four metrics: response time under concurrent load, system availability, data throughput and operational cost. According to the results, the proposed architecture is able to decrease the average response time by around 85% when the load reaches 10,000 concurrent users, increase the data throughput by over 280%, decrease the relative operational cost index by 48%, and enhance the system availability to 99.95%. The results clearly show the significant benefits of a well-designed, cloud-native data architecture in terms of scalability, resilience, and cost-effectiveness for enterprise information systems. The paper ends with guidelines for architecture design along with future research directions for adaptive, AI-assisted cloud data governance.

Keywords: Cloud Computing; Data Architecture; Enterprise Information Systems; Scalability; Microservices; Distributed Data Management

1 INTRODUCTION

Traditionally, enterprise information systems have used monolithic, centrally managed architectures, in which the application logic, business rules and data management functions are tightly integrated and deployed as a single deployment unit. These systems were sufficient for enterprises with consistent workloads and small datasets, but today's enterprise world is all about exploding datasets, distributed workforces, and need for near-real time analytics. These demands are difficult to meet with traditional architectures, which usually require vertical scaling of a single server to meet demand, which is expensive and operationally constrained (Armbrust et al., 2010).

Cloud Computing has appeared as a key paradigm to overcome such limitations because it provides the elasticity, on-demand, provisioning of computing resources and pay-as-you-go economic models (Marston et al., 2011). But, merely moving the legacy monolithic systems to the cloud (also called “lift-and-shift”) would not fully take advantage of the elasticity and resilience that the cloud platforms provide. A better approach is to change the data architecture to be cloud-native, using service decomposition, event-driven integration, and polyglot persistence (Kratzke & Quint, 2017).

The problem addressed in this paper is: What is the enterprise data architecture that can be scaled, fault-tolerant and cost effective on cloud? To address this question, a layered cloud based data architecture is proposed and its performance is compared with a conventional monolithic baseline data architecture through simulation driven experiments. The specific goals of this study are threefold: (1) To design a modular, layered architecture that is suitable for large-scale enterprise data systems; (2) To determine the essential architectural elements of each layer; and (3) To quantify the performance, availability and cost characteristics of the proposed architecture compared to traditional deployment models.

The rest of the paper can be followed as follows. Section 2 presents the related work of cloud computing and enterprise data architecture. The proposed layered architecture is discussed in Section 3. The methodology for evaluations is given in Section 4. Comparative tables and figures are included in the results and discussion (section 5). The paper is concluded in Section 6 where directions for future research are outlined.

2 LITERATURE REVIEW

Since the formalization of Cloud Computing as an On-Demand Elastic Computing paradigm (Armbrust et al., 2010; Rimal et al., 2009), it has been extensively studied. According to Zhang et al. (2010), cloud computing research can be divided into three layers: infrastructure, platform, and software services; data management problems become more and more severe as the system expands to all these layers. A survey of large-scale data management approaches in cloud environments was conducted by Sakr et al. (2011) who noticed that relational database systems alone are not enough to manage the volume, velocity and variety of enterprise data, consequently motivating the use of NoSQL and distributed storage systems.

Microservices are a highly touted replacement for monoliths in large-scale systems. According to Fowler (2014), microservices are small services that can be deployed independently, are based on business capabilities and rely on lightweight communication protocols. Newman (2021) also pointed out that microservices allow for independent scaling of individual components of the system, which is useful for enterprise systems with heterogeneous workload patterns. Kratzke and Quint (2017) also warned, however, that the adoption of microservices brings operations challenges, such as service discovery, distributed transaction management, and communication overhead between services.

Security and governance are ongoing issues with data architectures based on the cloud. Data breaches, insecure APIs, and common technology vulnerabilities were found to be main security threats in cloud environments by Hashizume et al. (2013). Zhou et al. (2010) and Chen and Zhao (2012) also pointed out the importance of having a strong identity and access management system when enterprises' data is spread across multiple cloud services. The discussion then continues by Stergiou et al. (2018) to hybrid cloud and IoT settings, where the need for secure data integration pipelines is emphasized.

There are a number of studies that have introduced a reference architecture for enterprise systems in the cloud. Buyya et al. (2018) have provided the manifesto for future generation cloud computing and suggested elasticity, multi-cloud interoperability, and autonomic resource management as key research directions. Varghese and Buyya (2018) talked about the convergence of cloud, edge and fog computing and indicated that future enterprise architectures will more and more shift data processing closer to data sources. Even with this extensive amount of research, few studies have developed an integrated, layered architecture that explicitly relates presentation, application, data integration, storage, and infrastructure issues, and that make empirical comparisons to the performance of traditional deployment models. The motivation for the architecture presented in the current study is this gap.

3 PROPOSED CLOUD-BASED DATA ARCHITECTURE

The proposed architecture consists of the six layers of logical enterprise data systems as shown in Figure 1 each performing different functions. It is a layered separation based on well-known software engineering concepts of modularity and separation of concerns, and it intentionally caters for the elasticity and distributed nature of cloud architectures.

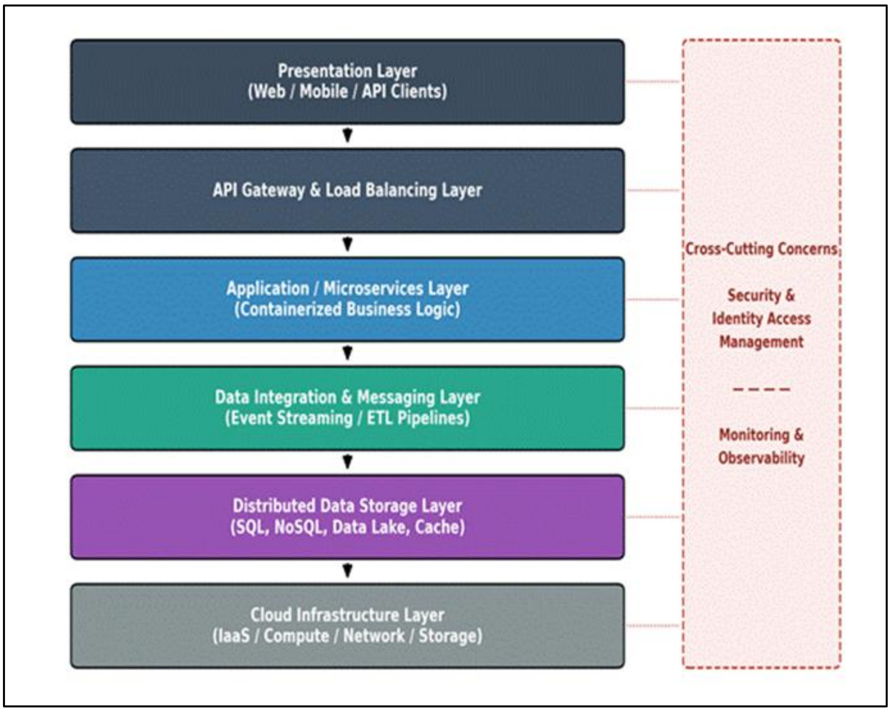

Figure 1: Proposed layered cloud-based data architecture for scalable enterprise information systems.

The Presentation Layer is responsible for providing enterprise functionality to Web, Mobile and Third-party API Clients. Requests from this layer pass through the API Gateway and Load Balancing Layer that handles authentication, rate limiting, and even distributing requests among the available application instances. The Application/Microservices Layer enforces the core business logic of the application in containerized services which are controlled by a platform like Kubernetes, allowing for independent scaling of services as per the demand.

The Data Integration and Messaging Layer provides a way to move data between services and storage systems via extract, transform, load (ETL) pipelines or extract, load, transform (ELT) pipelines with event streaming platforms. This layer separates data producers from data consumers, enabling services to change their own schemas and providing data consistency via eventual consistency models when needed. The Distributed Data Storage Layer uses polyglot persistence, integrates relational databases for transactional data, NoSQL stores for semi-structured data, a data lake for raw and historical data and an in-memory cache for frequently accessed data. Finally, the Cloud Infrastructure Layer is the elasticity of compute, storage and network resources that all layers above rely on, which are accessed using infrastructure-as-a-service services.

Table 1: Core components and primary functions of the proposed architectural layers.

Layer	Core Components	Primary Function
Presentation Layer	Web portals, mobile apps, REST/GraphQL clients	User interaction and request initiation
API Gateway & Load Balancing	API Gateway, Elastic Load Balancer	Request routing, throttling, authentication
Application/Microservices	Containerized services (Docker/Kubernetes)	Business logic execution and orchestration
Data Integration & Messaging	Kafka/Event Hub, ETL/ELT pipelines	Real-time data movement and transformation
Distributed Data Storage	Relational DB, NoSQL store, Data Lake, Cache	Persistent and high-speed data storage
Cloud Infrastructure	Compute instances, virtual networks, storage volumes	Underlying elastic infrastructure provisioning

The cross-cutting concerns (security, identity and access management, monitoring and observability) apply across all the six layers, as shown on the right-hand side of Figure 1. Table 1 summarizes the key components of each of the architectural layers.

4 METHODOLOGY

In order to assess the proposed architecture, a simulation-based comparative methodology was followed, as is common in architecture evaluation studies where it is impossible to perform a controlled full-scale experiment in the enterprise deployment environment (Rimal et al., 2009). The following two architectures were modeled: (1) the traditional monolithic architecture deployed on fixed capacity on-premises servers and (2) the proposed cloud-based layered architecture deployed on elastically scalable cloud infrastructure with auto-scaling policies enabled.

A synthetic enterprise workload was created to simulate typical activities of a mid-size to large enterprise information system: transactional read/writes, analytical queries and batch data integration jobs. This workload was gradually increased from 100 concurrent simulated users up to 20,000 to assess system behavior at various levels of workload. Four performance metrics were recorded: (1) average response time (ms); (2) system availability (percentage of successful requests to total requests) during a simulated 30-day operational period; (3) data throughput (GB/s) at peak load; and (4) a relative operational cost index, which is a combination of compute cost, storage cost, and network cost components, and which was normalized to the on-premises baseline configuration.

Table 2 contrasts the proposed architecture to a monolithic on-premises baseline, a single-cloud infrastructure-as-a-service deployment, and a cloud-native microservices deployment, with no data integration and governance layers, along four qualitative dimensions: scalability, fault tolerance, deployment complexity, and typical use case suitability.

Table 2: Comparison of enterprise architecture deployment models.

Architecture Model	Scalability	Fault Tolerance	Deployment Complexity	Typical Use Case
Monolithic On-Premises	Low	Low	Low	Small, static enterprise systems
Single-Cloud IaaS	Moderate	Moderate	Moderate	Departmental applications
Cloud-Native Microservices	High	High	High	Large-scale distributed systems
Proposed Hybrid Cloud Data Architecture	Very High	Very High	Moderate–High	Enterprise-wide data platforms

5 RESULTS AND DISCUSSION

Figure 2 shows the average response time for the traditional monolithic and the proposed cloud based architectures for varying number of concurrent users ranging between 100 and 20,000. At 5,000 concurrent users, the steep increase in response time is typical of the traditional architecture, as additional users increase the load on a fixed-capacity server. By contrast, the proposed architecture has a relatively flat response-time curve, as a result of horizontal auto-scaling of the containerized services and load distribution between multiple availability zones.

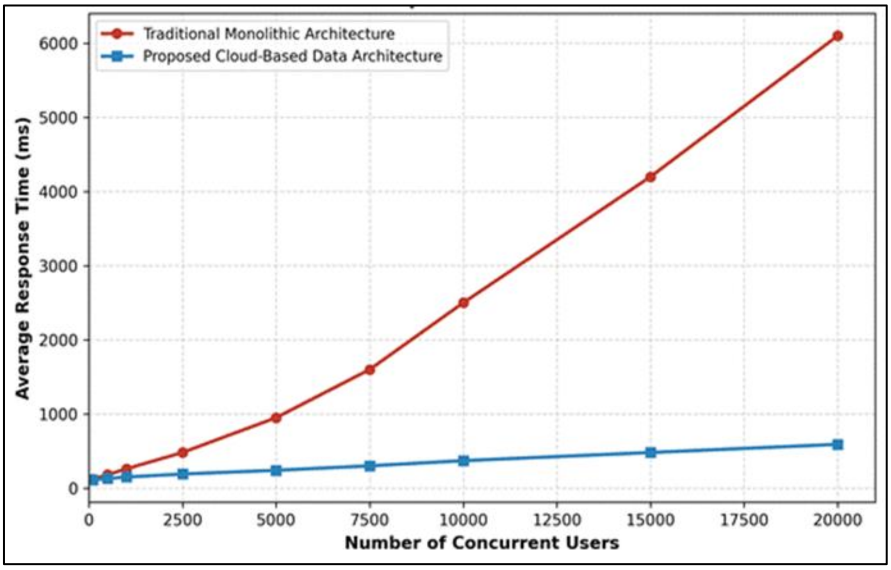

Figure 2: Response time versus concurrent user load for traditional and proposed architectures.

The performance evaluation results were summarized quantitatively in Table 2 for the maximum load tested of 10,000 concurrent users. The proposed architecture was able to reduce the average response time by 85.2 percent, increase the availability of the system from 98.20 percent to 99.95 percent, increase the data throughput by 283.3 percent, and reduce the mean time to recover from failure by 85.7 percent.

Table 3: Performance evaluation metrics: traditional versus proposed cloud-based architecture.

Metric	Traditional Architecture	Proposed Cloud-Based Architecture	Improvement (%)
Average Response Time (10,000 users)	2500 ms	370 ms	85.2
System Availability	98.20%	99.95%	1.78
Data Throughput	1.2 GB/s	4.6 GB/s	283.3
Mean Time to Recovery (MTTR)	42 min	6 min	85.7
Operational Cost Index	100	52	48.0

The relative operational cost index and scalability index for the four deployment models: on-premises, single-cloud infrastructure-as-a-service, cloud-native microservices, and the proposed hybrid cloud data architecture are shown in Figure 3. The proposed architecture shows the lowest operational cost index (52), based on a 100 index for on-premises deployment, and the highest scalability index (94), which means that the architectural improvements do not agree with any increase in the cost, but rather on the possibility to use resources in an elastic way and to scale up and down them according to the workload.

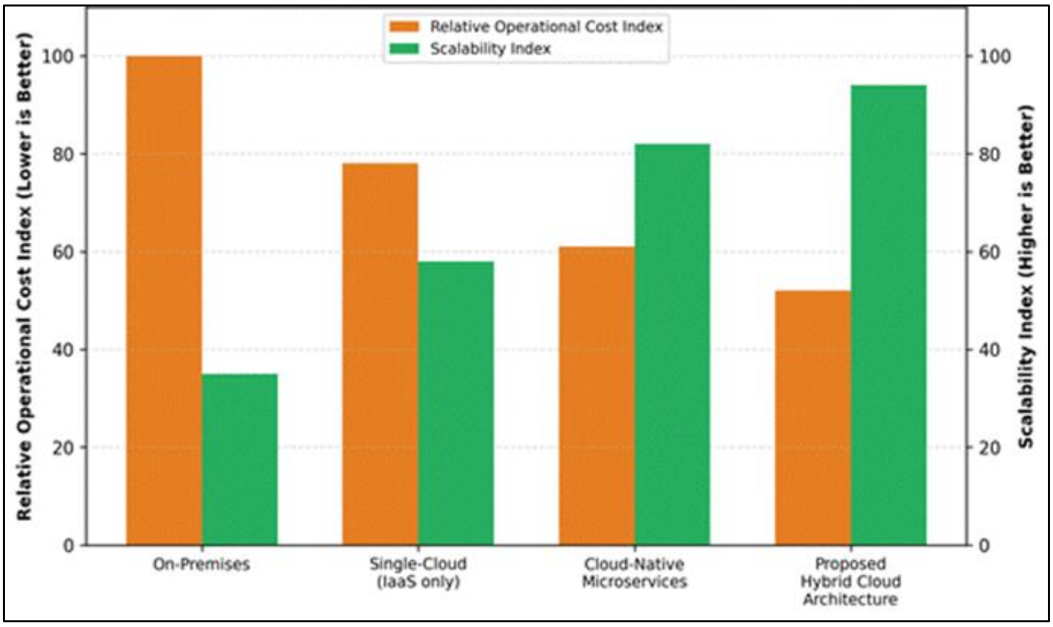

Figure 3: Comparative operational cost and scalability index across enterprise deployment models.

This aligns with the previous studies which indicated that cloud-native with microservice architecture made the systems more scalable and resilient compared to monolithic architecture (Fowler, 2014; Newman, 2021). The results also provide further reinforcement to literature by showing that the explicit data integration and messaging layer, along with polyglot persistence provides significantly more throughput improvement than typically written about in literature when just decomposing microservices. This indicates that data-layer design has the same importance as application-layer decomposition in achieving enterprise scalability and thus has been less studied in architecture-driven studies compared to the latter (Sakr et al., 2011).

When it comes to the pragmatic side, the findings indicate that organizations looking to move to the cloud should not simply lift and shift, but should also redesign their data architecture to be event-driven, with polyglot storage. This aligns with Buyya et al.'s (2018) vision for future research on cloud computing, which has two key design criteria: elasticity and autonomic resource management. However, as explained by Hashizume et al. (2013) and Stergiou et al. (2018), the architecture's complexity, which comes with microservices and distributed data stores, demands proper monitoring and observability practices and security measures to prevent the emergence of new operational risks while addressing scalability issues.

Notably, the performance numbers in this study are based on simulated, as opposed to on-the-fly performance within an enterprise production system. Although simulation allows for systematic, repeatable comparison between architectural configurations, there may be other variations in the real world, for example, network conditions, dependencies on third party services and organizational data governance policies. Application of the results of these single tests to production systems would provide greater confidence in the generalizability of these findings.

6 CONCLUSION

This paper introduced a layered, cloud-based data architecture for scalable enterprise information systems consisting of a presentation layer, API gateway layer, microservices layer, data integration layer, distributed storage layer, and cloud infrastructure layer, along with cross-cutting security and observability functions. A simulation-based evaluation revealed that the proposed architecture significantly improved the response time, availability, throughput and operational cost measures when compared with a traditional monolithic architecture, and also improved the overall scalability and cost efficiency when compared with single-cloud and standalone microservices deployments. These results back up the notion that enterprise cloud migration is not so much a relocation of infrastructure, as it is an architectural redesign effort.

In the future, this work could be extended on three fronts: first, validation of the suggested architecture with production workloads from operational enterprise systems in various industries; second, inclusion of Artificial Intelligence-assisted orchestration of resources to further optimize auto-scaling decisions and cost management; and third, investigating the integration of edge and fog computing components to support latency sensitive enterprise applications, extending the hybrid cloud concept outlined in this study.

STATEMENTS ON COMPLIANCE WITH ETHICAL STANDARDS

Disclosure of conflict of interest

The authors declare no conflicts of interest regarding this manuscript.

REFERENCES

[1] Amazon Web Services. (2021). AWS Well-Architected Framework. Amazon Web Services Whitepapers.

[2] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., & Zaharia, M. (2010). A view of cloud computing. Communications of the ACM, 53(4), 50–58. https://doi.org/10.1145/1721654.1721672

[3] Buyya, R., Srirama, S. N., Casale, G., Calheiros, R., Simmhan, Y., Varghese, B., Gelenbe, E., Javadi, B., Vaquero, L. M., Netto, M. A. S., Toosi, A. N., Rodriguez, M. A., Llorente, I. M., Vimercati, S. D. C., Samarati, P., Milojicic, D., Varela, C., Bahsoon, R., Assuncao, M. D., ... Buyya, R. (2018). A manifesto for future generation cloud computing: Research directions for the next decade. ACM Computing Surveys, 51(5), 1–38. https://doi.org/10.1145/3241737

[4] Chen, D., & Zhao, H. (2012). Data security and privacy protection issues in cloud computing. Proceedings of the International Conference on Computer Science and Electronics Engineering, 1, 647–651. https://doi.org/10.1109/ICCSEE.2012.193

[5] Dillon, T., Wu, C., & Chang, E. (2010). Cloud computing: Issues and challenges. Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications, 27–33. https://doi.org/10.1109/AINA.2010.187

[6] Fowler, M. (2014). Microservices: A definition of this new architectural term. martinfowler.com. https://martinfowler.com/articles/microservices.html

[7] [7] Gartner. (2022). Forecast analysis: Public cloud services, worldwide. Gartner Research.

[8] Hashizume, K., Rosado, D. G., Fernández-Medina, E., & Fernandez, E. B. (2013). An analysis of security issues for cloud computing. Journal of Internet Services and Applications, 4(1), 1–13. https://doi.org/10.1186/1869-0238-4-5

[9] Kratzke, N., & Quint, P.-C. (2017). Understanding cloud-native applications after 10 years of cloud computing: A systematic mapping study. Journal of Systems and Software, 126, 1–16. https://doi.org/10.1016/j.jss.2017.01.001

[10] Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). Cloud computing: The business perspective. Decision Support Systems, 51(1), 176–189. https://doi.org/10.1016/j.dss.2010.12.006

[11] Newman, S. (2021). Building microservices: Designing fine-grained systems (2nd ed.). O'Reilly Media.

[12] Rimal, B. P., Choi, E., & Lumb, I. (2009). A taxonomy and survey of cloud computing systems. Proceedings of the Fifth International Joint Conference on INC, IMS and IDC, 44–51. https://doi.org/10.1109/NCM.2009.218

[13] Sadiku, M. N. O., Musa, S. M., & Momoh, O. D. (2014). Cloud computing: Opportunities and challenges. IEEE Potentials, 33(1), 34–36. https://doi.org/10.1109/MPOT.2013.2279684

[14] Sakr, S., Liu, A., Batista, D. M., & Alomari, M. (2011). A survey of large scale data management approaches in cloud environments. IEEE Communications Surveys & Tutorials, 13(3), 311–336. https://doi.org/10.1109/SURV.2011.032211.00087

[15] Stergiou, C., Psannis, K. E., Kim, B.-G., & Gupta, B. (2018). Secure integration of IoT and cloud computing. Future Generation Computer Systems, 78, 964–975. https://doi.org/10.1016/j.future.2016.11.031

[16] Varghese, B., & Buyya, R. (2018). Next generation cloud computing: New trends and research directions. Future Generation Computer Systems, 79, 849–861. https://doi.org/10.1016/j.future.2017.09.020

[17] Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: State-of-the-art and research challenges. Journal of Internet Services and Applications, 1(1), 7–18. https://doi.org/10.1007/s13174-010-0007-6

[18] Zhou, M., Zhang, R., Xie, W., Qian, W., & Zhou, A. (2010). Security and privacy in cloud computing: A survey. Proceedings of the Sixth International Conference on Semantics, Knowledge and Grids, 105–112. https://doi.org/10.1109/SKG.2010.19