Scholars Journal of Engineering and Technology
Abbreviated Key Title: Sch J Eng Tech
ISSN 2347-9523 (Print) | ISSN 2321-435X (Online)
Journal homepage: https://saspublishers.com

OPEN ACCESS

A Serverless, Event-Driven Architecture for High-Performance Background Check Order Creation

Shrikant Dnyandev Pawar[1*], Mahesh Barre[2]

[1]Senior Manager, Software Engineering, First Advantage, 1 Concourse Parkway NE, Suite 200, Atlanta, GA 30328.
[2]Infrastructure Team Lead, Digital Technology and Innovation, Platform Engineering and Shared Services, Wells Fargo, 1525 WT Harris Blvd., Charlotte, NC 28262.

DOI: https://doi.org/10.36347/sjet.2025.v13i10.006 | **Received:** 28.08.2025 | **Accepted:** 16.10.2025 | **Published:** 24.10.2025

***Corresponding author:** Shrikant Dnyandev Pawar
Senior Manager, Software Engineering, First Advantage, 1 Concourse Parkway NE, Suite 200, Atlanta, GA 30328.

Abstract **Original Research article**

The background check industry continues to face inefficiencies in manual order creation, resulting in long turnaround times and high risks of human error. These challenges are especially critical in recruitment, financial services, and compliance-driven workflows where speed, accuracy, and regulatory adherence are essential. Manual processes average ~180 seconds per request and are difficult and costly to scale. This work presents a serverless, event-driven system for automated order creation that leverages functions-as-a-service (FaaS) and event queues to improve latency, scalability, and reliability while lowering operational costs (Kaffes, K). The design allows for independent scaling of individual components based on demand, ensuring responsiveness during peak loads (Exton, K). Initial results demonstrate a substantial reduction in order creation latency compared to traditional methods, alongside enhanced fault tolerance and cost-effectiveness (Chard, R). This approach provides a blueprint for modernizing background check processes and can be adapted to other data-intensive applications requiring high throughput and availability (Meister, B). The architecture integrates AWS Lambda, Step Functions, DynamoDB, SQS, and API Gateway to handle real-time requests and orchestrate distributed workflows with built-in retries, error handling, and operational alerts for greater resilience. Experimental results show a reduction in average latency from ~180 to ~13 seconds (over 90%), a 95% cost reduction, and reliability improvements from ~70% to ~98%. These outcomes demonstrate the potential of serverless architectures to transform compliance-driven, data-intensive operations through scalable, efficient, and sustainable automation.

Keywords: Serverless, Event-Driven Architecture, Background Checks, AWS Lambda, Workflow Automation, Compliance Systems, Scalability, High-Performance Computing.

I. INTRODUCTION

Background check order creation is a critical, yet often cumbersome process across various sectors, including recruitment, financial services, and regulatory compliance (Boldachev, A) (Kaffes, K). Traditional methods often rely on manual data entry, email exchanges, and disparate systems, leading to inefficiencies, errors, and delays. These manual processes are not only time-consuming but also introduce significant risks related to data security and compliance with evolving regulations (Meister, B).

Serverless architecture offers a compelling solution to address these challenges. By leveraging event-driven principles, a serverless approach enables the creation of highly scalable, cost-effective, and resilient background check order creation workflows. This paradigm shifts the focus from managing infrastructure to defining business logic, allowing organizations to respond quickly to fluctuating demands and optimize resource utilization.

This work introduces a serverless, event-driven architecture design to optimize the background check order creation process. The system leverages the benefits of function-as-a-service (FaaS) computing to automate data processing, validation, and integration with external background check providers.

II. BACKGROUND AND RELATED WORK

A. Evolution of Application Architectures

Business process automation has progressed from monolithic to microservices architectures. While monolithic systems are simple to build and deploy for small-scale solutions, they struggle with scalability, maintainability, and adaptability to evolving business needs (Lazzari, L). Even minor updates often require full redeployment, increasing downtime and risk.

Microservices addressed these issues by decomposing applications into independently deployable components that communicate over a network (Meister, B). This improves scalability and fault isolation but introduces new challenges in distributed communication, transaction management, and orchestration complexity. Managing dependencies and monitoring numerous services can offset some of these advantages (Lazzari, L).

B. The Rise of Serverless and Event-Driven Computing

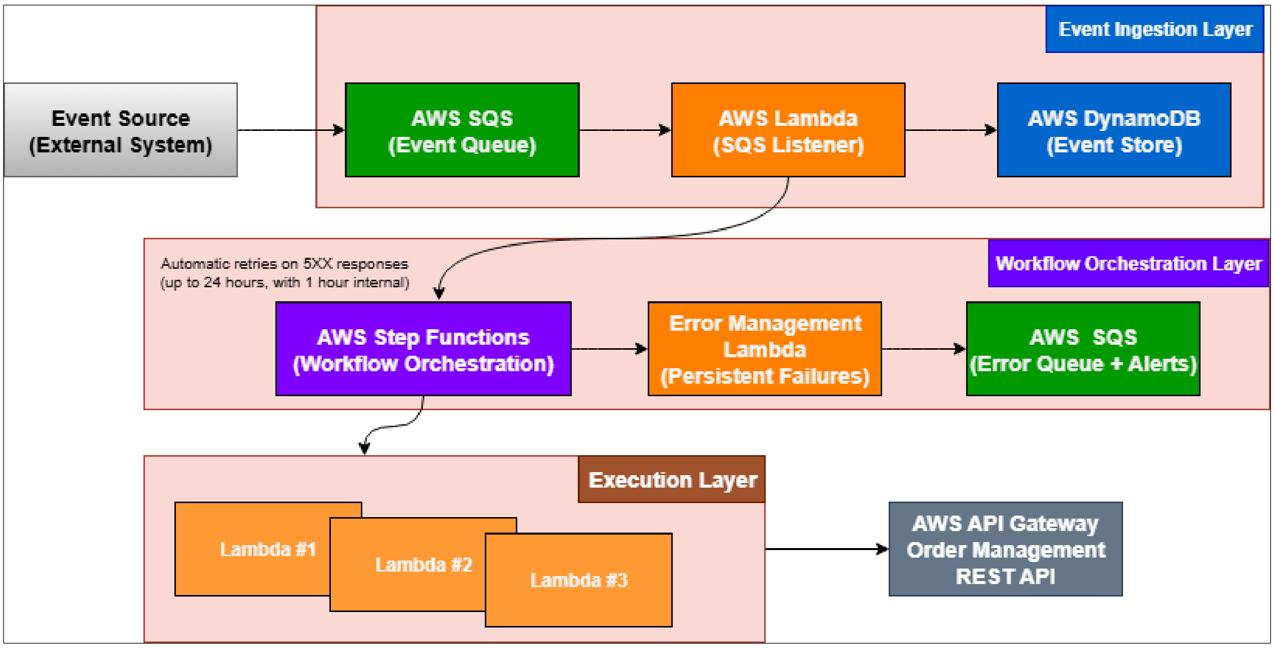
Serverless computing represents the next evolution, abstracting infrastructure management and enabling developers to focus on event-driven logic (Boldachev, A). These systems are inherently scalable, elastic, and cost-efficient, well-suited for asynchronous, high-volume workloads. Workflow orchestration frameworks extend serverless systems by modeling complex processes as interconnected tasks, simplifying automation across distributed services. This approach has proven effective in domains such as logistics, healthcare, and manufacturing (Lazzari, L), demonstrating its flexibility and robustness for large-scale, data-intensive operations.

C. Gap in Current Research

Despite these advances, few studies explore serverless architectures for background check automation. This domain presents distinct challenges: integrating diverse data sources, maintaining regulatory compliance, and ensuring high decision accuracy. The lack of focused research highlights the need for scalable, compliant and precise automation frameworks for background check workflows.

III. SYSTEM ARCHITECTURE AND DESIGN

This section outlines a serverless, event-driven architecture designed for high-performance background check order creation. The system leverages the scalability and cost efficiency of serverless computing to process large request volumes with low latency and high availability. It follows three-layer architecture-Event Ingestion, Workflow Orchestration, and Execution. While conceptually cloud-agnostic, the system is implemented using AWS services including Lambda, Step Functions, SQS, DynamoDB, and API Gateway. The architecture promotes loose coupling, independent scalability, and real-time responsiveness by activating resources only when needed. The high-level architecture is illustrated in Fig. 1.



(Fig. 1. High-level system architecture depicting the event-driven pipeline.)

A. Layered Architecture and Processing Flow

- **Event Ingestion Layer:** Incoming order requests queued up in SQS through upstream applications and consumed by an AWS Lambda listener. The request data persisted in DynamoDB for state tracking, ensuring idempotent retries and eliminating manual initiation delays.
- **Workflow Orchestration Layer:** Employs a serverless orchestration service (AWS Step Functions) to coordinate the sequence of tasks for order creation, managing state transitions, error handling, and overall workflow control. This replaces the manual sequencing of steps with automated state management.

- **Execution Layer:** Implements stateless serverless functions responsible for background check order creation tasks such as input validation and record generation via API calls. By decomposing the workflow into granular, stateless functions, the architecture eliminates manual bottlenecks, improves fault isolation, and enhances maintainability. Resource allocation and scaling are automatically managed by the serverless platform, ensuring efficiency and resilience.

This layered, event-driven design ensures scalability, resilience, and adaptability. Each component operates independently yet cohesively, enabling efficient, real-time order creation without idle resource consumption.

B. Data Management and State Persistence

DynamoDB serves as the primary data store for order request payloads and order state management throughout the creation process. The initial order request persisted in DynamoDB, with state updated progressively as the workflow advances. This design ensures idempotency, enabling safe retries without duplicate order creation, while providing state recovery capabilities during failures. DynamoDB's inherent scalability and high availability make it ideally suited for handling large volumes of concurrent order creation requests while avoiding Step Functions' payload size limitations.

C. Reliability and Error Handling

The system implements a comprehensive, multi-layered error-handling strategy. Transient errors are first managed through configurable retry policies within Step Functions, enabling automatic recovery from temporary failures. When retries are exhausted, executions transition to a dedicated error-handling Lambda function that captures detailed failure context and stores the event in an SQS queue for later analysis. CloudWatch continuously monitors the SQS queue and related error metrics; when predefined thresholds are exceeded, it triggers an SNS or email alert to notify operators. This layered approach ensures that errors are promptly detected, contained, and resolved while maintaining overall system integrity.

D. System-Wide Optimization and Governance

- **Performance Optimization:** The architecture employs short-lived, stateless Lambda functions to minimize cold start latency, while parallel execution in Step Functions accelerates workflow completion. Strategic memory allocation and function sizing balance performance with cost efficiency, enabling the dramatic reduction from ~180 seconds (manual) to ~13 seconds (automated) processing time.
- **Security and Compliance:** Comprehensive security measures include fine-grained IAM policies, KMS-managed encryption for data at rest and in transit, and TLS-secured communications. Regular security audits and penetration testing address vulnerabilities, ensuring regulatory compliance and protecting sensitive applicant data from unauthorized access (Li, C).
- **Cost Efficiency:** The serverless model delivers economic advantages through pay-per-use pricing, charging only for actual Lambda executions, Step Functions transitions, and DynamoDB storage. With no upfront costs or idle charges, combined with auto-scaling that aligns capacity with demand, the system maintains high performance while optimizing resource utilization and minimizing operational expenses.

E. Illustrative Use Cases and Scenarios

This section demonstrates practical applications of serverless architecture through key scenarios:

- **High-Volume Processing** For organizations processing thousands of daily background checks, the architecture automatically scales to handle demand without manual intervention. Orders are queued and processed in parallel by dynamically scaled Lambda functions, ensuring timely completion regardless of volume fluctuations.
- **Multi-Provider Integration** The system seamlessly integrates with multiple background check providers through dedicated Lambda functions. When an order is submitted, the architecture determines required providers, invokes corresponding functions, and aggregates results for unified presentation.
- **Real-Time Monitoring and Alerting** The event-driven design enables real-time monitoring with automatic alerts for delayed orders or processing errors. This allows rapid issue identification and resolution, maintaining workflow efficiency and order completion timelines.
- **Regulatory Compliance** The architecture supports compliance with regulations like FCRA through built-in security controls and audit logging. Automated data protection measures, including sensitive information redaction in logs and reports, ensure ongoing compliance and privacy preservation.

IV. EXPERIMENTAL SETUP AND EVALUATION

The effectiveness of the proposed system was evaluated through controlled experiments against a manual baseline. The analysis focused on four key metrics- latency, throughput, reliability, and cost efficiency.

A. Test Environment

The automated system was deployed on Amazon Web Services (AWS) using a suite of serverless services. AWS Lambda handled core processing tasks, triggered by events from Amazon SQS (Simple Queue Service). Data persistence was managed with Amazon DynamoDB, selected for its low latency and scalability. The infrastructure was provisioned via AWS CloudFormation to enforce infrastructure-as-code and ensure repeatability. IAM roles were configured with least-privilege permissions across all components to maintain security and isolation.

B. Baseline System (Manual Process)

The baseline for comparison was a manual workflow where operators entered candidate data from emails or web forms into a legacy system. Each request averaged ~180 seconds to process, with variations based on workload and operator efficiency. Throughput scaled linearly with staffing, and the process was prone to data entry errors.

C. Proposed Serverless System

The serverless system automates the background check order creation pipeline end to end. When a new request enters the SQS queue, an AWS Lambda function validates the data, transforms it for the provider's API, initiates the background check, and updates the order status in DynamoDB after each execution. This ensures real-time state tracking and consistency throughout the workflow. Leveraging stateless execution and elastic scaling, the system reduces average processing time from ~180 seconds to ~13 seconds per order while maintaining performance under high concurrency.

D. Workload and Metrics

The manual baseline was measured by timing multiple operator-driven order completions. The automated system was evaluated by submitting varying volumes of simulated requests to the pipeline. Metrics were consistently defined as follows:

- **Latency:** End-to-end order completion time.
- **Cost:** Operator labor cost for the manual process versus cloud resource usage cost for the automated system.
- **Throughput:** Number of orders completed per minute.
- **Reliability:** Percentage of orders successfully processed without manual intervention.

V. RESULTS

The serverless, event-driven architecture delivered significant performance gains over the manual system. Evaluated across latency, cost, throughput, and reliability, the results demonstrate substantial improvements in operational efficiency and scalability, positioning serverless as a viable solution for modernizing background checks in a landscape demanding faster turnaround and greater resilience (Exton, K).

A. Latency Reduction

Manual order creation averaged ~180 seconds per request. The automated workflow reduced this latency to ~13 seconds, representing an improvement of over 90%. The automated process also exhibited significantly lower variance, leading to more consistent and predictable compliance outcomes (**see Fig. 2a**).

B. Cost Efficiency

The serverless architecture reduced processing costs by over 95%, from approximately $15 to $0.35 per 1,000 orders. Unlike manual costs that scale linearly with staffing and infrastructure, the serverless pay-per-use model provides predictable, workload-based pricing. This eliminates idle resource costs, delivering substantial financial benefits, especially at high volumes (**see Fig. 2b**).

C. Throughput

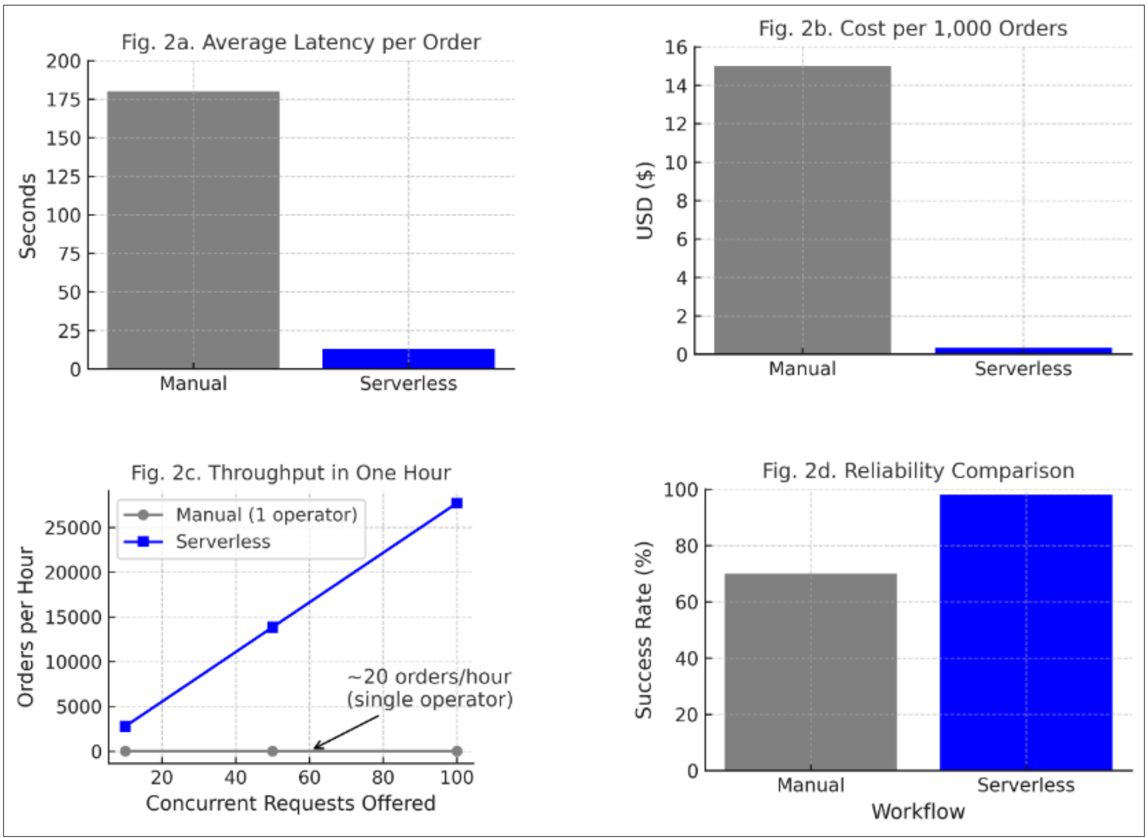
Manual throughput averaged 0.33 orders per minute per operator (~20 per hour), creating bottlenecks during peak periods. In contrast, the serverless system's event-driven design scaled automatically with demand, sustaining thousands of concurrent orders up to AWS service limits. **(see Fig. 2c)**.

D. Reliability

The manual workflow achieved a success rate of approximately 70% due to human error and process variability. The serverless execution, benefiting from automated retries, validation, and error handling, achieved a success rate of ~98% (**see Fig. 2d**).

Table I: Performance comparison: manual vs. serverless workflow

Metric	Manual Workflow	Serverless Workflow	Improvement
Latency	~180 s per order	~13 s per order	>90% faster
Cost	~$15 per 1,000 orders	~$0.35 per 1,000 orders	>95% reduction
Throughput	~0.33 orders/min (1 operator) ~20/hr	Thousands of orders/hr (auto-scaled)	Near-linear scaling
Reliability	~70% success rate	~98% success rate	+28% absolute



(Fig. 2. Evaluation results: (a) Latency per order (seconds), (b) Cost per 1000 orders (USD), (c) Throughput - orders per hour, (d) Reliability - success rate (%))

E. Summary of Results

As shown in Table I, the serverless, event-driven system reduced latency by a factor of 14, scaled elastically to handle high concurrency, reduced costs by over 95%, and improved absolute reliability by 28 percentage points. These results demonstrate the ability of serverless computing to modernize background check operations, combining speed, scalability, efficiency, and reliability while lowering operational overhead.

VI. LIMITATIONS AND FUTURE WORK

While the evaluation confirms significant advantages in scalability and cost-efficiency, several limitations must be acknowledged. This section outlines these challenges and key directions for future research.

A. Limitations

- **Cold Start Latency:** AWS Lambda functions may experience initialization delays (cold starts) of several hundred milliseconds after periods of inactivity, which can accumulate in workflows with multiple sequential functions.
- **State Management Constraints:** AWS Step Functions impose a 256 KB payload limit and a one-year execution maximum. The stateless nature of serverless functions often necessitates external state stores, adding complexity (Li, C).
- **Vendor Lock-In:** The architecture's deep integration with proprietary AWS services complicates and increases the cost of migration to other cloud providers.
- **Cost Dynamics:** While highly cost-efficient for bursty traffic, the pay-per-execution model can lead to high and unpredictable costs for consistently high-frequency workloads without careful monitoring.

B. Future Research Directions

- **Enhanced State Management:** Research into lightweight frameworks or "durable functions" that natively manage state could simplify orchestration and reduce external dependencies.
- **Improved Portability:** Developing abstraction layers or adopting open-source serverless frameworks would mitigate vendor lock-in and enable practical multi-cloud deployments.
- **Domain Extensions:** The architecture could be extended to include integrated compliance checks, candidate data enrichment, and advanced reporting pipelines to broaden its applicability.
- **Advanced Cost Forecasting:** Creating accurate cost-prediction models that incorporate execution time, memory allocation, and workload patterns is essential for predictable budgeting in high-volume scenarios.

VII. SECURITY AND COMPLIANCE IMPACT

The adoption of a serverless, event-driven architecture has significantly strengthened the security and compliance posture of the background screening system. Automated workflows, executed through AWS Step Functions and Lambda, ensure deterministic processing with complete audit trails, thereby enhancing adherence to regulations such as GDPR and FCRA and reducing risks of non-compliance.

Security has been improved through the application of least-privilege IAM policies, KMS-backed encryption of data at rest, and TLS protection of data in transit. These measures safeguard sensitive personal information, mitigate exposure to breaches, and support GDPR requirements for data protection. Error-handling and automated retries enhance resilience, minimizing risks of silent data loss and compliance violations. Workflow decomposition into stateless functions reduces the attack surface and improves fault isolation. Immutable execution histories and encrypted DynamoDB storage provide verifiable audit records, supporting GDPR subject-rights requests and FCRA dispute resolution obligations. Automation also reduces human error in data handling, improving accuracy in line with FCRA's fairness requirements. The architecture's scalability ensures consistent compliance performance under high-volume workloads, reinforcing GDPR principles of accountability and integrity.

Overall, the serverless design delivers a robust, auditable, and compliant background screening system that balances stringent regulatory obligations with high operational efficiency and reliability.

VIII. ADVANTAGES AND BROADER IMPACT

The serverless, event-driven architecture for high-performance background check order creation delivers benefits across technical, organizational, and societal dimensions. By combining scalability, agility, and resilience with fairness, transparency, and sustainability, it offers a transformative model for compliance-driven processes.

A. Scalability, Efficiency, and Adaptation

Serverless and event-driven systems inherently enable rapid scaling and efficient resource utilization. Concurrency and dynamic load balancing allow the system to absorb demand surges without performance loss. Each event triggers a dedicated function, supporting parallel execution that eliminates bottlenecks and accelerates order turnaround. Modularity and decoupling enhance resilience and simplify updates, while rapid deployment cycles enable adaptation to evolving business and regulatory needs. Traditional architectures, by contrast, often struggle to scale and adapt efficiently (Chard, R).

B. Organizational Advantages: Efficiency and Security

Automation improves efficiency and reduces costs by minimizing manual intervention and streamlining compliance reporting (Chard, R). The serverless model eliminates infrastructure overhead, with pay-per-use pricing aligning expenses directly with workload. Security is reinforced through fine-grained IAM controls, encryption, and automatic scaling safeguards—crucial for protecting sensitive personal data. Real-time monitoring strengthens resilience against emerging threats. Together, these capabilities enable organizations to process higher volumes with greater accuracy and fewer resources while maintaining compliance with GDPR and FCRA.

C. Societal and Humanitarian Impact: Fairness, Transparency, and Sustainability

Automation reduces human error and bias, promoting fairness and consistency in decisions across employment, finance, and other regulated sectors. Transparent audit trails enhance accountability and public trust. Beyond commercial use, the scalability of serverless computing supports humanitarian efforts such as volunteer vetting during disaster relief, where speed and reliability are critical. By executing only when triggered, serverless systems minimize idle resources and energy consumption, advancing environmental sustainability and aligning with global carbon-reduction goals.

IX. CONCLUSION

This paper presented the design and evaluation of a serverless, event-driven architecture for high-performance background check order creation. The system, implemented with AWS Lambda and Step Functions, addressed the inefficiencies of manual workflows by reducing average latency from ~180 seconds to ~13 seconds, an improvement of over 90%. Experimental results further demonstrate near-linear scalability, high reliability (~98% success rate), and significant cost savings (>95% reduction) through a pay-per-use model.

By leveraging the elasticity and event-driven nature of serverless computing, the architecture enables real-time responsiveness, seamless integration, and greater agility compared to traditional monolithic systems. These findings underscore the transformative potential of serverless approaches in compliance-driven domains, providing organizations with a scalable, cost-efficient, and resilient foundation for background check processing.

While challenges such as cold start latency, orchestration constraints, and vendor lock-in remain, they also highlight clear directions for future research. Overall, this work demonstrates that serverless computing is not only a viable but also a strategically advantageous solution for modernizing compliance-critical processes.

ACKNOWLEDGEMENTS

I would like to thank Sushil Mishra, Lead Quality Automation Engineer, First Advantage for his valuable insights and guidance during the preparation of this paper.

REFERENCES

- Kaffes, K., Yadwadkar, N. J., & Kozyrakis, C. (2021). Serverless computing has seen rapid growth due to the ease-of-use and cost efficiency it provides. arXiv. https://arxiv.org/abs/2111.07226
- Exton, K., & Read, M. (2024). To support parallelizable serverless workflows in applications like media processing, we have prototyped a distributed scheduler called Raptor. arXiv. https://arxiv.org/abs/2403.16457
- Chard, R., Skluzacek, T. J., Li, Z., Babuji, Y., Woodard, A., Blaiszik, B., ..., & Chard, K. (2019). Growing data volumes and velocities are driving exciting new methods across the sciences. arXiv. https://arxiv.org/abs/1908.04907
- Meister, B., Baskaran, M., Pradelle, B., Henretty, T., & Lethin, R. (2016). As illustrated by the emergence of a class of new languages and runtimes, it is expected that a large portion of the programs to run on extreme scale computers. arXiv. https://arxiv.org/abs/1601.05458
- Boldachev, A. (2025). This paper presents boldsea, Boldachev's semantic-event approach -- an architecture for modeling complex dynamic systems. arXiv. https://arxiv.org/abs/2509.09775
- Lazzari, L., & Farias, K. (2023). Event-driven architecture has been widely adopted in the software industry. arXiv. https://arxiv.org/abs/2308.05270
- Li, C., Huang, L., He, D., Wen, Y., Liu, G., & Duan, L. (2025). The serverless platform aims to facilitate cloud applications' straightforward deployment, scaling, and management. arXiv. https://arxiv.org/abs/2503.06532