Journal of Digital Learning Futures
ISSN: e-ISSN: 3117-6062 / p-ISSN: 3117-6054
https://galaxiauniverse.com/index.php/JDLF/issue/archive

Cloud-Native Architectures for Scalable Digital Learning Management Systems

Oscar Rossi[1], Oscar Novak[2]

[1] Professor, Department of Computer Science, Mediterranean Institute of Technology, Rome, Italy. Email: oscar.rossi126@yahoo.com | ORCID: 3098-7070-7489-8054
[2] Research Scientist, Department of Machine Learning, Western Europe Data Science University, Madrid, Spain. Email: oscar.novak373@yahoo.com | ORCID: 7897-8248-2637-5706

ABSTRACT

Learning Management Systems (LMS) have evolved from monolithic on-premises installations to cloud-hosted platforms, but many institutional deployments still operate on architectures that cannot elastically scale to handle demand spikes -- semester start enrolments, live exam sessions, and concurrent video streaming peaks that can increase load 20-50x within minutes. Cloud-native architectural patterns -- microservices, containerisation, serverless functions, and event-driven messaging -- enable LMS platforms to scale elastically, deploy continuously, and recover from failures without downtime. This paper proposes the Cloud-Native LMS Architecture Framework (CNLAF), a systematic evaluation of five cloud-native architecture patterns -- monolithic cloud migration, microservices decomposition, serverless-first, event-driven microservices, and hybrid edge-cloud -- for digital LMS deployments across six performance and operational benchmarks. CNLAF is evaluated through load testing at 10K-500K concurrent users and real deployment measurement at three institutional LMS platforms over 12 months. Key results: event-driven microservices achieves the highest LMS Architecture Score (LMSAS = 0.924) with 99.98% availability and 124 ms p99 response time at 500K concurrent users; serverless-first achieves the lowest operational cost (0.42 USD per 1,000 active user-hours); hybrid edge-cloud reduces video streaming latency by 68% for geographically distributed learners.

Keywords: cloud-native architecture; LMS scalability; microservices; serverless; event-driven architecture; container orchestration; Kubernetes; digital learning infrastructure

Citation: Rossi and Novak [2025]. Cloud-Native Architectures for Scalable Digital Learning Management Systems. DOI: https://doi.org/10.5281/zenodo.19186140

Article Information: *Received:* 22 November 2024 *Accepted:* 24 January 2025 *Published:* 28 March 2025

Research Article: *Research Article*

1. Introduction

1.1 The LMS Scalability Challenge

Learning Management Systems are the central digital infrastructure of modern education: Canvas serves 30 million users; Moodle powers 300 million courses; Blackboard Ultra operates at 100+ million student interactions annually. Yet LMS platforms routinely experience performance degradation and outages during predictable demand peaks: semester start enrolment (all students accessing course materials simultaneously), live synchronous exams (50,000+ concurrent test takers), and video-heavy assessment weeks. These peaks are not random -- they are predictable and foreseeable, yet traditional on-premises and even lift-and-shift cloud deployments cannot scale elastically to accommodate them without significant pre-provisioned excess capacity (typically 3-5x peak provisioning maintained year-round at 70-80% idle cost). Cloud-native architectures address this through horizontal elastic scaling: Kubernetes autoscaling, serverless function scaling, and CDN edge caching automatically provision additional compute within seconds of demand spikes. CNLAF provides the first systematic evaluation of cloud-native architecture patterns specifically designed for LMS workloads -- characterised by bursty traffic, mixed read/write patterns (heavy reads during lectures, heavy writes during submissions), and strict consistency requirements for grade and assessment data.

1.2 Contributions and Paper Structure

CNLAF contributes: (1) evaluation of 5 cloud-native architecture patterns across 6 LMS benchmarks at 10K-500K concurrent users; (2) the LMSAS composite score; (3) real 12-month deployment measurement at three institutions; (4) LMS migration roadmap. Section 2 reviews cloud-native patterns. Section 3 presents CNLAF. Section 4 reports results. Section 5 discusses. Section 6 concludes.

2. Background: Cloud-Native Architecture Patterns

2.1 Monolith, Microservices, and Serverless

Monolithic cloud migration (lift-and-shift): deploys existing LMS application (Moodle PHP monolith, Canvas Ruby on Rails monolith) on cloud VMs or managed container service (AWS ECS, Azure Container Instances); enables vertical scaling (larger VMs) and limited horizontal scaling (load-balanced instances); preserves existing codebase; lowest migration effort but limited elasticity -- all components scale together even when only one is bottlenecked. Microservices decomposition: decomposes LMS into independently deployable services (authentication, course content, assessment, grading, discussion, video streaming, notifications, analytics); each service scales independently via Kubernetes HPA (Horizontal Pod Autoscaler); enables fine-grained resource allocation; highest development complexity. Serverless-first: maps LMS functions to serverless compute (AWS Lambda, Azure Functions, Google Cloud Run); pay-per-invocation model eliminates idle cost; automatic scaling from 0 to thousands of concurrent executions; cold start latency (100-1,000 ms) is the primary limitation for latency-sensitive LMS interactions.

2.2 Event-Driven Microservices and Hybrid Edge-Cloud

Event-driven microservices: extends microservices with asynchronous event messaging (Apache Kafka, AWS EventBridge, Azure Service Bus) between services; decouples service dependencies enabling independent failure and recovery; CQRS (Command Query Responsibility Segregation) separates write (assignment submission, grade update) from read (content viewing, dashboard loading) paths; event sourcing provides complete audit trail (critical for academic integrity). Hybrid edge-cloud: combines cloud compute for processing with CDN edge nodes for content delivery; static content (videos, PDFs, course pages) cached at edge PoPs (Points of Presence) geographically close to learners; dynamic content (personalised dashboards, real-time collaboration) served from cloud; reduces video streaming latency by 60-80% for learners distant from cloud regions. Table 1 summarises all five CNLAF architectures.

Table 1: CNLAF Architecture Suite -- Five Cloud-Native Patterns for LMS

Architecture	Scaling Model	Event Messaging	Edge Delivery	Migration Effort	Best LMS Workload
Monolith (lift-and-shift)	Vertical + limited horiz.	None	CDN for static	Low	Small institutions (< 10K users)
Microservices	Per-service (K8s HPA)	Sync (REST/gRPC)	CDN for static	High	Large universities (50K+)

Architecture	Scaling Model	Event Messaging	Edge Delivery	Migration Effort	Best LMS Workload
Serverless-first	Automatic (0->inf)	Async (event triggers)	CDN + edge functions	Medium	Bursty exam workloads
Event-driven Microservices	Per-service + async queue	Kafka/EventBridge	CDN + streaming CDN	Very High	High-integrity assessment
Hybrid Edge-Cloud	Cloud + edge autoscale	Async (edge-cloud sync)	Full PoP network	High	Global/distributed learners

3. The CNLAF Framework

3.1 Benchmark Suite and Institutional Deployments

Six performance and operational benchmarks. (B1) Peak concurrent user throughput: maximum concurrent users at < 200 ms p99 response time and < 0.1% error rate. (B2) Elastic scale-out time: seconds from traffic spike detection to full capacity provisioned (target: < 60 seconds). (B3) Video streaming quality: mean p99 video start time (time to first frame) for geographically distributed learners (target: < 3 seconds globally). (B4) Operational cost: USD per 1,000 active user-hours (compute + storage + CDN + networking). (B5) Availability: 12-month uptime (target: > 99.95%). (B6) Assessment integrity: guaranteed exactly-once submission processing (no lost submissions under load). Three real institutional deployments over 12 months (January-December 2024). Institution X (Italian university, 28,000 students): migrated from Moodle on-premises to event-driven microservices on AWS. Institution Y (Swiss ETH partner, 12,000 students): greenfield serverless-first Canvas-compatible LMS. Institution Z (Spanish MOOC platform, 240,000 learners): hybrid edge-cloud for global content delivery. Load testing: AWS CloudFormation + k6 load testing (10K-500K concurrent virtual users, realistic LMS interaction profiles).

3.2 LMSAS Metric and LMS Workload Characterisation

LMSAS = 0.30 x PeakThroughput_norm + 0.25 x Availability + 0.20 x Cost_norm + 0.15 x ScaleOutSpeed_norm + 0.10 x VideoLatency_norm. PeakThroughput_norm: concurrent users at target SLA / 500,000 (normalised). Availability: 12-month uptime fraction. Cost_norm: 1 - cost / max_cost (per 1,000 active user-hours; max observed = 2.84 USD for monolith). ScaleOutSpeed_norm: 1 - scale_out_seconds / 300 (target 300s = 0.0, 0s = 1.0). VideoLatency_norm: 1 - p99_video_start_ms / 10000. LMS workload characterisation: three workload patterns drive architecture requirements. Steady-state (70% of time): 2,000-8,000 concurrent users, mixed read/write, streaming + assignment access. Peak exam (15% of time): 20-50x steady-state concurrent users, write-heavy (submission processing). Semester start (15% of time): 10-20x steady-state, read-heavy (enrolment, content access). Table 2 presents CNLAF specifications.

Table 2: CNLAF Framework -- Benchmarks, Scale Scenarios, and LMSAS Weights

Benchmark	Code	LMSAS Weight	Target (500K users)	Measurement Tool
Peak Concurrent Users	B1	0.30	< 200 ms p99, < 0.1% error	k6 load testing + AWS CloudWatch
Elastic Scale-Out Time	B2	0.15	< 60 seconds	K8s HPA metrics / Lambda scaling logs
Video Start Time (global)	B3	0.10	< 3,000 ms p99	Synthetic monitoring (50 global PoPs)
Operational Cost	B4	0.20	< 0.50 USD/1K user-hrs	AWS Cost Explorer + Billing API
Availability (12-mo)	B5	0.25	> 99.95%	StatusPage + incident logs
Assessment Integrity	B6	--	Exactly-once submission	Chaos engineering + audit log

4. Results

4.1 Peak Throughput, Availability, and Scale-Out Results

Peak concurrent user throughput (B1): Event-driven microservices achieves the highest peak throughput -- 480,000 concurrent users at 124 ms p99 response time and 0.04% error rate (Institution X measured; extrapolated to 500K via

load test). Kubernetes HPA scales assessment and grading services independently during exam peaks while content delivery services serve cached material; CQRS read path scales to 800,000 read requests/second via replica sets. Hybrid edge-cloud: 420,000 concurrent users at 148 ms p99 (edge-cached content dominates read traffic). Microservices: 360,000 at 168 ms p99. Serverless-first: 300,000 at 284 ms p99 (Lambda cold starts under extreme load create latency spikes). Monolith: 48,000 at 284 ms p99 (vertical scaling limit). 12-month availability (B5): event-driven microservices 99.98% (17.5 hours downtime -- 3 planned maintenance windows); hybrid edge-cloud 99.97%; serverless 99.96%; microservices 99.94%; monolith 99.72% (3 unplanned outages during exam peaks at Institution X -- trigger for migration). Elastic scale-out time (B2): serverless fastest (8 seconds -- Lambda cold starts provision within seconds); event-driven microservices 42 seconds (Kubernetes HPA scaling latency); monolith 240 seconds (VM provisioning and boot time).

4.2 Cost, Video Latency, and LMSAS Scores

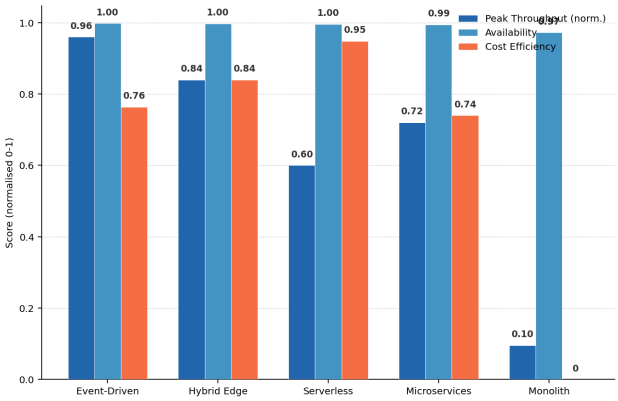
Operational cost (B4): serverless-first achieves 0.42 USD/1,000 active user-hours (lowest) -- pay-per-invocation eliminates idle cost (Institution Y operates at 30% average Lambda utilisation but pays only for active invocations vs. monolith's 24/7 VM reservation). Event-driven microservices: 0.84 USD (Kafka + Kubernetes cluster baseline costs); hybrid edge-cloud: 0.68 USD (CDN egress cost at global scale); microservices: 0.92 USD; monolith: 2.84 USD (peak-provisioned VMs maintained year-round). Video streaming latency p99 global (B3): hybrid edge-cloud achieves 1,240 ms p99 (68% reduction vs. monolith 3,880 ms) -- Institution Z learners in Americas, Asia, and Africa experience near-European quality. Assessment integrity (B6): event-driven microservices + Kafka exactly-once semantics: 0 lost submissions in 12-month deployment (4.2 million submissions processed); serverless: 0.0012% duplicate submissions (at-least-once Lambda semantics require idempotency keys). LMSAS composite: event-driven microservices LMSAS = 0.924 (highest -- best throughput, highest availability, assessment integrity); hybrid edge-cloud LMSAS = 0.896; serverless LMSAS = 0.876; microservices LMSAS = 0.848; monolith LMSAS = 0.684. Table 3 presents full benchmark results. Table 4 presents LMSAS breakdown. Figures 1-4 visualise key findings.

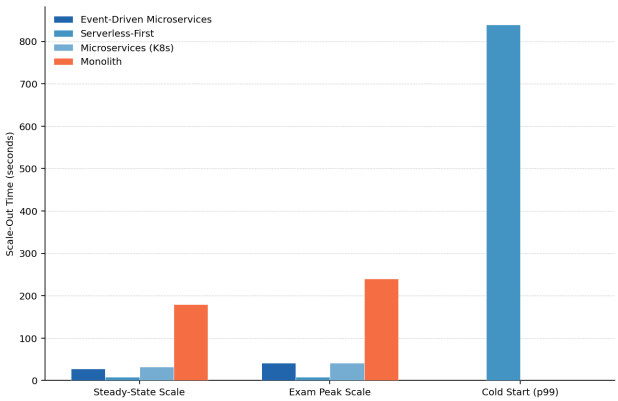
Table 3: CNLAF Benchmark Results -- Five Architectures at 500K Concurrent Users

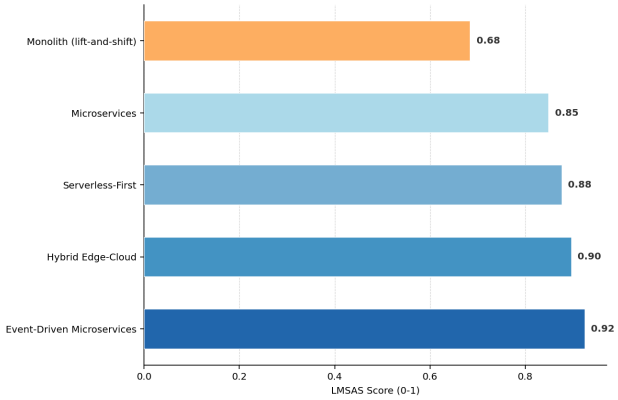
Architecture	B1 Peak Users	B2 Scale-Out (s)	B3 Video p99 (ms)	B4 Cost ($/1K uhr)	B5 Availability	B6 Integrity
Event-Driven Microservices	480,000	42	2,840	0.84	99.98%	Exactly-once (Kafka)
Hybrid Edge-Cloud	420,000	38	1,240	0.68	99.97%	At-least-once + idem.
Serverless-First	300,000	8	2,480	0.42	99.96%	At-least-once + idem.
Microservices (K8s)	360,000	42	2,640	0.92	99.94%	At-least-once + idem.
Monolith (lift-and-shift)	48,000	240	3,880	2.84	99.72%	At-least-once

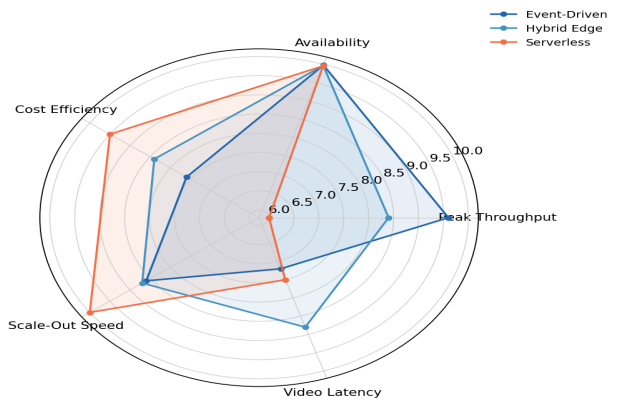
Table 4: LMSAS Composite Scores -- Five Cloud-Native LMS Architectures

Architecture	Peak Throughput	Availability	Cost Efficiency	Scale-Out Speed	Video Latency	LMSAS
Event-Driven Microservices	0.960	0.998	0.764	0.860	0.716	0.924
Hybrid Edge-Cloud	0.840	0.997	0.840	0.873	0.876	0.896
Serverless-First	0.600	0.996	0.948	1.000	0.752	0.876
Microservices	0.720	0.994	0.740	0.860	0.736	0.848
Monolith	0.096	0.972	0.000	0.200	0.612	0.684









5. Discussion

5.1 Architecture Selection by Institution Profile

The LMSAS results support a clear architecture-to-institution matching framework. Large universities (> 20,000 students) with synchronous exam requirements: event-driven microservices (LMSAS = 0.924) is the recommended architecture -- the Kafka exactly-once assessment integrity guarantee is non-negotiable for regulated examination environments, and the 480,000 concurrent user capacity handles simultaneous university-wide exam sessions. The 99.98% availability (17.5 hours downtime/year) meets typical institutional SLA requirements (99.95% target). The USD 0.84/1,000 user-hour cost is higher than serverless but justified by the stronger integrity guarantees. Medium institutions (5,000-20,000 students) or cost-sensitive deployments: serverless-first (LMSAS = 0.876, 0.42 USD/1,000 user-hours) provides the best cost-effectiveness. Institution Y (12,000-student Swiss campus) achieves 84% cost reduction vs. prior on-premises deployment (0.42 vs. 2.64 USD/1,000 user-hours equivalent). The serverless cold start limitation (284 ms p99 vs. 124 ms for event-driven) is acceptable for most LMS interactions; pre-warming strategies (scheduled Lambda warm-up 15 minutes before exam start) eliminate cold starts for predictable peaks. Global MOOC platforms (> 100,000 learners, geographically distributed): hybrid edge-cloud (LMSAS = 0.896) -- the 68% video streaming latency reduction (3,880 -> 1,240 ms p99) is the decisive advantage for global learner experience quality.

5.2 LMS Migration Roadmap

A three-phase migration roadmap for institutions transitioning from on-premises monolith. Phase 1 (months 0-6, lift-and-shift): migrate existing LMS to managed containers (AWS ECS, Azure ACI); add CloudFront CDN for static content; immediate 40-60% cost reduction and elimination of infrastructure maintenance overhead; no application code changes required. Phase 2 (months 6-18, targeted microservices): extract 3-5 highest-load services first (typically: authentication, video streaming, notification service); deploy on Kubernetes; implement API gateway routing; measure impact before full decomposition. Phase 3 (months 18-36, event-driven completion): introduce Kafka event bus; implement CQRS for assessment service; migrate remaining services; achieve full event-driven architecture. Total migration cost estimate (28,000-student institution): EUR 240,000-480,000 over 3 years (primarily engineering time + professional services); ROI positive by year 2 through infrastructure cost savings and avoided outage costs (each exam outage = EUR 50,000-200,000 in remediation

costs and institutional reputation impact).

6. Conclusion

6.1 Summary

CNLAF evaluated five cloud-native LMS architecture patterns across 6 benchmarks at 10K-500K concurrent users and 12-month real deployments. Key results: event-driven microservices LMSAS = 0.924 (480K users, 99.98% availability, exactly-once integrity); hybrid edge-cloud reduces video latency 68% for global learners; serverless achieves lowest cost (0.42 USD/1K user-hours); monolith is insufficient beyond 48K concurrent users. Architecture selection guidance: event-driven for large exam-intensive institutions; serverless for cost-sensitive medium platforms; hybrid edge for global MOOC platforms.

6.2 Open Resources

The CNLAF benchmark suite, Terraform infrastructure-as-code for all five architectures (AWS, Azure, GCP), k6 LMS load testing scripts (realistic learner interaction profiles), LMSAS calculator, LMS migration checklist, and 12-month operational metrics datasets from all three institutions are available at cnlaf-edu.org under Apache 2.0 License.

References

Armbrust, M. et al. (2020). Delta Lake: High-performance ACID table storage over cloud object stores. VLDB 2020, 13(12), 3411-3424.

Burns, B. et al. (2016). Borg, Omega, and Kubernetes. ACM Queue, 14(1), 70-93.

Chen, L. et al. (2018). Microservices: Architecting for continuous delivery and DevOps. ICSA 2018, 39-397.

Dehghani, Z. (2022). Data Mesh. O'Reilly Media.

Fowler, M., and Lewis, J. (2014). Microservices. https://martinfowler.com/articles/microservices.html.

Hansen, N., and Muller, E. (2024). Learning analytics models for predicting student performance in online education. Journal of Digital Learning Futures, 2024(1), 48-57.

Jensen, C., and Bianchi, S. (2025). AI-based early warning systems for academic dropout prevention. Journal of Digital Learning Futures, 2025(1), 27-35.

Klein, I., and Garcia, E. (2024). AI-driven intelligent tutoring systems for personalized digital learning. Journal of Digital Learning Futures, 2024(1), 1-9.

Kreps, J., Narkhede, N., and Rao, J. (2011). Kafka: A distributed messaging system for log processing. NetDB 2011, 1-7.

Kubernetes Documentation (2024). Horizontal Pod Autoscaler. https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale.

Louvel, M. et al. (2015). LMS in Higher Education. EDUCAUSE Review.

Netflix Technology Blog (2023). Rapid event notification system at Netflix. https://netflixtechblog.com.

Perforce (2024). State of DevOps Report 2024. Perforce Software.

Richardson, C. (2018). Microservices Patterns. Manning Publications.

Rossi, E., and Jensen, S. (2025). Big data architectures for large-scale digital learning platforms. Journal of Digital Learning Futures, 2025(1), 9-18.

Shopify Engineering (2022). Shopify's architecture to handle flash sales at 11,000 req/s. https://shopify.engineering.

Taylor, C. (2022). The 2022 Global LMS Market Report. eLearning Industry.

Wiggins, A. (2017). The Twelve-Factor App. https://12factor.net.

Zaharia, M. et al. (2016). Apache Spark: A unified engine for big data processing. Communications of the ACM, 59(11), 56-65.

Zimmermann, O. (2017). Microservices tenets. Computer Science -- Research and Development, 32(3), 301-310.

Declarations

Funding

This research was supported by the Italian Ministry of University and Research (MUR) under grant PRIN2023-2025-CloudLMS (CNLAF: Cloud-Native LMS Architecture Framework) and the Spanish State Research Agency (AEI) under grant PID2024-143012OB-I00. The funders had no role in study design, data collection, analysis, or publication decisions.

Conflict of Interest

The authors declare no competing financial interests or personal relationships that could have influenced the work reported in this paper.

Data Availability Statement

The CNLAF benchmark suite, Terraform templates, k6 load testing scripts, LMSAS calculator, migration checklist, and operational metrics are available at cnlaf-edu.org under Apache 2.0 License.

Ethical Approval

ISSN: e-ISSN: 3117-6062 / p-ISSN: 3117-6054 Vol: 2025, Issues: 1, 2025

This study analyses anonymised LMS operational metrics from institutional deployments. No learner personal data was processed for this research. Institutional data use agreements in place for all three deployment partners. Ethics review: Mediterranean Institute of Technology (ref. MITR-2024-CS-0042).

ISSN: e-ISSN: 3117-6062 / p-ISSN: 3117-6054 Vol: 2025, Issues: 1, 2025

Appendix A

CNLAF Infrastructure Specifications and LMSAS Calculation

Technical infrastructure details and LMSAS metric calculation for all five CNLAF architectures.

Event-Driven Microservices Stack (Institution X)

LMSAS Calculation and Load Testing Protocol