Scholar Publishing Group
Machine Learning Theory and Practice
https://doi.org/10.38007/ML.2026.060110
ISSN 2790-0983 Vol. 6, Issue 1: 85-93

Research on Design and Implementation of a Distributed Anomaly Detection System for Financial Logs in High-Concurrency Risk Control and Settlement Links

Ziyu Zhang

Sichuan Normal University, Chengdu, 610000, China

Keywords: Financial logs; Distributed anomaly detection; Streaming computation; Log semantic representation; AIOps

Abstract: Financial institutions continuously generate high-dimensional, heterogeneous and time-series log data in various stages of transaction matching, risk control decision-making, clearing and reconciliation, as well as infrastructure operation and maintenance. Traditional single-machine log analysis solutions have major deficiencies in high-throughput and low-latency operation, cross-node correlation detection, stability under template drift and label scarcity, and false alarm control. This paper presents a distributed anomaly detection system framework of "collection-parsing-representation-detection-interpretation-backflow" for financial log scenarios. Kafka and Flink are employed for streaming access and state computation. A semantic representation, sequence model and rule constraint fusion mechanism are introduced at the model layer. Feature caching, window aggregation, online threshold calibration and alarm grading are used at the engineering level to improve the real-time performance and interpretability of the system. Based on research results and benchmark data in the field of log anomaly detection over the past three years, a methodological analysis will be conducted, and a system design strategy suitable for the collaborative monitoring of financial business logs, system logs, and security logs will be put forward. Research has shown that a distributed processing architecture and multi-source feature fusion models can simultaneously achieve the goals of high-throughput scalability, high-accuracy detection, and easy-to-maintain deployment; thus, a practical path for anomaly detection technology has been proposed for securities, payment, banking and quantitative trading platforms.

1 Introduction

Algorithmic trading, securities matching, payment clearing, real-time risk control systems, etc., are all moving towards cloud-native and distributed architectures, and the logs produced by financial institutions are showing an upward trend in frequency, continuous accumulation, and cross-domain coupling. Transaction gateway logs contain order access status, queue congestion and reasons for rejected orders; risk control logs record rule triggering, limit control, and behavioural scoring; and database and messaging system logs handle node status, retry arrangements and

consistency failures. Compared with the operating environment of the Internet, financial log anomaly detection also needs to identify whether the abnormality is due to a system failure or a risk in business operations, fund security and other compliance issues. Therefore, the detection system needs to perform the aggregation, contextual correlation and anomaly-level classification of various source logs in a very short period of time.

Research in the past three years has shown that log anomaly detection has moved away from template matching and simple statistical thresholds for offline analysis to incorporating semantic representation, sequence learning, cross-domain transfer, and interpretable output in online intelligent analysis. Researchers also believe that anomalies in public log datasets do not have to be strong sequences; the features of the dataset, the parsing method and the representation format will all affect the results [1][2][4]. Therefore, general models cannot be directly applied to the system design in a financial context; instead, a cooperative relationship among the four components must be established: data organisation, distributed processing framework, detection model, and interpretation feedback loop.

The three contents of this paper are: first, the difficulties in distributed deployment of financial log data; second, some insights from existing research on real-time detection, cross-node correlation and explainable operations and maintenance; and third, how to develop a system that meets the requirements of accuracy, high throughput, low latency and easy maintenance. The contributions of this paper are to put forward a layered detection architecture suitable for financial business from a systems engineering perspective, to provide methods for multi-source feature fusion and online threshold calibration, and to summarize implementation approaches for practical deployment based on English literature and publicly available data from the past three years.

2. Current Status Analysis of the Research Topic

The four stages of existing research on log anomaly detection are generally classified as rule statistics, traditional machine learning, deep learning, and large model enhancement. Empirical studies have shown that deep learning is not necessarily superior to classical machine learning on some classic public datasets, and no stable improvements have been achieved after complex preprocessing [1]. Therefore, given the task constraints at an early stage of system construction, a model that is too complex should not be used. Log representation methods are also used by many different detectors, and different detectors often work best with different types of log representations. Bag-of-words, template sequences, parameter preservation and context embedding all have certain applicable ranges.

For the development of the model, LogSD uses self-supervised learning and frequency masking to improve the learning of low-frequency important log patterns, and is suitable for cases with few labels but many normal samples [3]. LogLead reduces the difficulty of benchmark reproduction by using unified data loading, enhancement and detection methods, and demonstrates the value of engineering pipeline standardisation for log analysis research [4]. RAPID puts forward the idea of training immune retrieval enhancement, and, using pre-trained natural language processing models and token-level information, achieves real-time detection. Reduce training costs in high-frequency business scenarios to enhance the value of online deployment [5].

Around 2025, research started to pay attention to cross-domain generalisation, interpretability and cloud-environment adaptability. LogFormer enhanced the performance of cross-data domain generalization by introducing pre-training, adapter tuning and a Log-Attention mechanism, and achieved higher F1 scores and reduced training and testing time on the HDFS, BGL and Thunderbird datasets [6]. ADAMAS started with cloud service systems and, using domain adaptation and AutoML, improved the adaptability of models in a production environment [7].

MIDLog used a multi-instance learning approach for the imprecise labelling case and reduced the cost of experts annotating logs individually [8]. Financial engineering is also moving towards the double goal of early warning and in-depth analysis. At FinanSE 2025, JPMorgan Chase put forward a statistical model-based way to detect anomalies in financial system logs and believed that the analysis results help determine whether a financial business is feasible [9].

At the same time, as shown in the system overview, large language models can transform log analysis from the original classification model into an all-encompassing approach of understanding, generation and interpretation. However, there are still problems with computational cost, illusion risk and online inference stability. Therefore, the system solution for financial logs should follow the general approach of "distributed stream processing as the foundation, semantic models as the core, and rule verification and business interpretation as guarantees".

Table 1 shows the core data features and engineering impact of the distributed anomaly detection system for financial logs.

Dimension	Typical source	Engineering impact	Suggested handling
Volume	Gateway, OMS, DB, MQ	Burst writes and backpressure	Partitioned ingestion + buffering
Heterogeneity	Business, infra, security logs	Feature mismatch across services	Unified schema + semantic encoding
Temporal coupling	Order, risk, clearing chain	Long-span causal propagation	Windowed correlation + state store
Label scarcity	Manual incident tickets	Weak supervision and drift	Pseudo labels + feedback loop

Table 1 shows the difference between financial log anomaly detection and general operational logs. The demands of the financial scenario are relatively high: a stable output under a large log volume, rapid cross-link propagation, and the presence of missing labels. Therefore, the data layer needs to support a unified schema mapping and partitioned access; the computation layer should have state management and window association capabilities; and the model layer needs to consider weakly supervised learning and online feedback. As shown in the table, the first need for system development is to solve engineering problems.

3. Pose Questions

Based on the above research and the shortcomings of the financial scenario, the four issues can be summarised as follows. First, there is a conflict between the throughput and latency of the data collection and computation stages. Financial logs often occur at the same time as the peaks of opening, closing, settlement, and market fluctuations. If a centralised processing chain is employed, queue congestion will readily occur in log parsing, feature concatenation and model inference. Second, there is a cross-domain heterogeneity problem of data semantics. The field structures of order, risk control, database, container and network logs are very different. If a general-purpose template is employed, the business context will be lost. Third, the detection goal is not to identify a single type of abnormality, but rather to simultaneously detect three kinds of defects: system failures, operational issues, and regulatory breaches, and is highly susceptible to false alarms. Fourth, banks can provide an explanation for the alert that is traceable; simply outputting an anomaly score is not enough.

Most of the publicly available research has been conducted in ideal conditions (good data preprocessing, offline training). However, log template drift, field additions, service expansions and

changes in the type of anomaly are all very frequent in real-world financial platforms. If the system does not have online calibration and feedback loop functions, its performance will drop rapidly after a few days, even if it starts out accurate. Therefore, this paper believes that the main problem is not selecting a single best model, but rather how to build a distributed anomaly detection system that can support model updates, rule additions and feedback mechanisms.

$$z(i,t) = (x(i,t) - \mu(i)) / \sigma(i) \tag{1}$$

Equation (1) is the result of standardising the continuous monitoring index. It is the i-th observation value of the index at time t, and its mean and standard deviation are the lower and upper values of the stable interval, respectively. Standardisation can reduce the differences in the dimensions of the features and is therefore suitable for the following unified model.

$$e(t) = Emb(log(t)) + Pos(t) \tag{2}$$

Equation (2) shows the way of log text representation. Emb(log(t)) is the log semantic vector obtained by the log semantic vector, and Pos(t) is introduced through time location information. Template number alone does not show the reason for the financial log abnormality. Preserve the semantic information of the retained object and show the differences in business rule hits, interface timeouts, etc.

$$h(t) = F\_att(Q(t), K(t), V(t)) \tag{3}$$

Equation (3) shows the process of context modelling with an attention mechanism. Q, K and V are the query vector, key vector and value vector, respectively. Therefore, one need only consider all the log records that occur within the same-day window, determine if there is any pattern that cannot be discerned from these records alone, and so on.

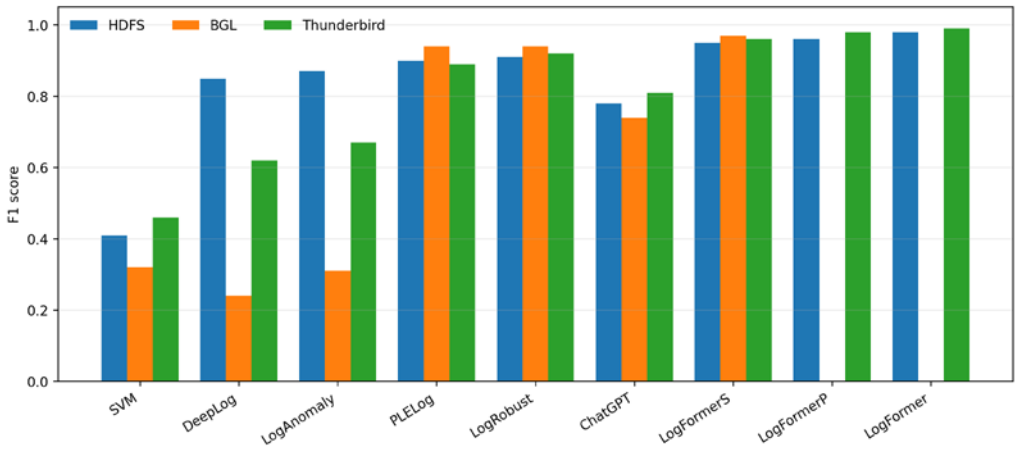

Figure 1 shows the comparison of F1 scores for several log anomaly detection methods using public benchmark datasets.

Figure 1 is a reconstruction based on the three publicly available results from the LogFormer paper: HDFS, BGL and Thunderbird. As shown in the table, traditional SVM performs poorly on all three datasets. However, by introducing semantic modelling and cross-domain adaptation, the LogFormer series of methods has achieved a higher overall F1 score and excels more in complex multi-domain logs. It can be seen that only shallow statistics or a single template matching method will not be suitable for addressing cross-business and cross-node anomaly patterns in finance. A Distributed Detection Framework Combining Semantics and Context is More Practical.

4. Problem Solving/Strategies

A five-layer, two-loop design pattern is proposed in this paper for the architecture of a distributed anomaly detection system for financial log data. The fifth layer is composed of a log access layer, a unified parsing layer, a feature and state layer, a detection and interpretation layer, and an alarm feedback layer; two closed loops have been formed: an online detection loop and an offline update loop. Kafka is used in the access layer for partitioned writing of multi-source logs, and these logs are divided into topics according to business domains, system domains, and security domains. The parsing layer has a common schema, template extraction and field standardisation, and produces computable events. Flink is used to store sliding windows, session contexts, and cross-node states in the feature and state layers. The three types of detection modules are semantic models, sequence models and rule engines. The feedback layer will save the results of alarm handling, manual judgment and back-tracking analysis in the sample database for use in threshold adjustment and model update.

This paper proposes that financial logs should be decomposed into four kinds of signals: semantic signals (log text embedding, key field encoding and event templates); temporal signals (sliding window frequency, time interval, state transition and session duration); topological signals (service call chains, node roles and upstream/downstream dependencies); and business signals (order status, account type, risk control tags and transaction stage). Use a multi-source signal fusion method to differentiate between "random alarms at the system level" and "significant anomalies affecting business results".

The model can be realised by using a light-weight online model combined with an improved offline model. The online path is low-latency, uses retrieval-based or lightweight neural models, and can quickly return anomaly scores and alarm levels. More complex sequence models, adapter fine-tuning and multi-instance learning methods are used in the offline path to relearn difficult samples and new anomalies [5, 6, 8]. A Design Approach should be adopted that does not place all complex models directly in the real-time link but continuously improves the system's robustness through offline incremental training.

$$s(t) = \lambda 1 \cdot d\_rec(t) + \lambda 2 \cdot d\_seq(t) + \lambda 3 \cdot d\_ctx(t) \quad (4)$$

Equation (4) is the total anomaly score function. d_rec is the reconstruction error or retrieval distance, d_seq is the sequence deviation, and d_ctx is the context conflict; respectively, λ_1, λ_2, and λ_3 are the weight coefficients. This Design can map the outputs of various detectors into a single scoring space and facilitate the real-time sorting and classification of alarms.

$$J = \alpha \cdot F1 + \beta \cdot Recall - \gamma \cdot P99Latency \quad (5)$$

Equation (5) is the optimisation objective of the system. Financial log anomaly detection should consider other than just a single F1 score, such as recall and P99 tail latency. Parameters α, β, and γ are employed to adjust the desired accuracy and speed requirements of the business. When the system is used for high-frequency trading or payment clearing link deployment, the penalty coefficient for tail latency is generally reduced.

Table 2 shows that the modules of data collection and feedback in the system are modular. The core of this design is the separation of parsing, state management, detection and feedback, and thus it will be easier to replace or add new rule bases later according to changes in the business. The modular structure of the system will reduce modification costs and can be used by multiple divisions of the bank. After adding a feedback console to the main process, alarm results will no longer be limited to the initial stage of anomaly detection but will trigger threshold adjustments, sample updates and knowledge base updates instead.

Table 2. Module Design of a Distributed Anomaly Detection System for Financial Logs

Layer	Key module	Main technique	Output
Ingestion	Collector + Kafka	Topic partitioning, batching	Ordered event stream
Parsing	Template + schema mapper	Regex, parser, field normalization	Structured log event
State	Flink state store	Window join, session context	Feature vector
Detection	Hybrid detector	Retrieval, attention, rules	Score + anomaly type
Feedback	Analyst console	Labeling, threshold tuning	Updated knowledge base

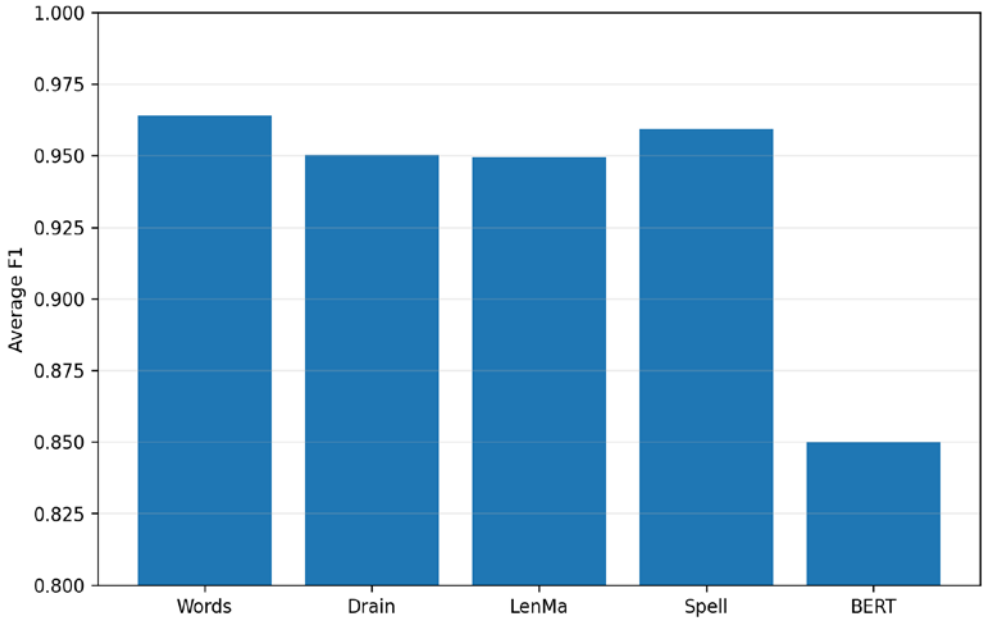

Figure 2. Average F1 Comparison of Different Log Representation Methods (Redrawn)

Figure 2 is a redrawn graph based on the results of LogLead published in a subset of HDFS. The average F1 scores of the term-level representations, Drain, LenMa, and Spell were all close to each other, and the BERT embedding did not achieve a higher average score in this experiment. Therefore, it is not necessary to use more complex methods for all log anomaly detection tasks. For the financial system, a reasonable method is to use multiple scenarios: computationally simple and easy-to-maintain representations for real-time processing, and more semantically complex models for verification and supplementation of offline processing.

Set the three thresholds in this paper to function as the real-time deployment. The first kind is a general-purpose statistical threshold that can detect abnormal increases in throughput, surges in error codes, or instability of latency. The second type is a model threshold, and it is used to adjust the anomaly scoring result. The third kind is a business threshold, and it considers factors such as funds, users, products and transaction periods to determine the severity of the alarm. The three types of thresholds work together to ensure that the system can maintain good alarm quality in all conditions, including opening auctions, nighttime batch processing and low load during the holiday.

The four divisions of abnormal output in terms of interpretation and generation are: triggering

event, relevant context, affected objects, and suggested actions. Information about the congestion of the transaction channel can be provided at the same time, such as log template drift, a rise in upstream retries, and an increase in downstream database lock waits. In case of an abnormal situation in the risk control rules, the relevant accounts, rule groups and similar previous events can be output, along with possible verification methods. It will help to improve the operating efficiency of the audit and meet the higher demand for work in the financial sector.

For system scalability, the distributed anomaly detection platform should also have features for hot model updates, feature version management and tenant-level isolation. The first function can switch model versions of different business areas smoothly, and the second will keep the governance boundaries of multiple teams and all business scenarios stable. Recent research trends show that in the future, the systems will increasingly use combinations of weak supervision, cross-domain migration and enhanced interpretation to adapt to changes in the data format of financial logs.

5. Summary

The first three parts of this paper are, respectively, the present situation of research, the abstraction of the problem, and system strategy. Research shows that the difficulty of financial log anomaly detection lies in many places: it is not only a lack of model recognition ability but also poor coordination among all these components, such as high-concurrency access, cross-domain semantic unification, online threshold calibration and interpretation feedback loops. This paper presents a complete system framework for practical deployment that uses Kafka and Flink as the streaming foundation, a unified schema and multi-source feature fusion for data, and lightweight online detection combined with offline reinforcement learning. Formulaic Goals for Assessment Accuracy and Real-time Performance are also provided.

According to the results in the literature and public benchmarks, log representation, training methods and cross-domain adaptation all affect the detection performance, but complex models are not necessarily better than simpler models [1][4][6]. Therefore, the building of the anomaly detection platform by financial institutions should follow the principles of "scenario priority, system priority and closed-loop priority", that is to say, first establish stable business data and computing platforms, and then introduce interpretable, transferable and updatable models in accordance with the alarm requirements. Subsequent work in this paper can continue to explore the creation of financial-specific datasets, multimodal early warning fusion, and model-assisted interpretation, etc., to enhance the reliability and practicality of the system.

References

[1] *Huang, J. (2025, August). Research on Multi-Model Fusion Machine Learning Demand Intelligent Forecasting System in Cloud Computing Environment. In 2025 2nd International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS) (pp. 1-7). IEEE.*

[2] *Hou, Y. (2026). Research on Heterogeneous Server Upgrade Strategies and Resource Utilization Efficiency Oriented Toward Green Computing Objectives. Advances in Computer and Communication, 7(1).*

[3] *Huang, J. (2025, September). Performance Evaluation Index System and Engineering Best Practice of Production-Level Time Series Machine Learning System. In 2025 International Conference on Intelligent Communication Networks and Computational Techniques (ICICNCT) (pp. 01-07). IEEE.*

[4] Hou, Y. (2026). Exploration of Data Center Cost Optimization Pathways Under Multi-generation CPU and GPU Collaborative Architectures. Engineering Advances, 6(1).

[5] Yanchun Wang. (2025) Research on Enhancing ERP System Efficiency through AI in Cross-border Supply Chain Environments. Advances in Computer and Communication, 6(5), 268-273.

[6] Zhang, C., Han, J., Zou, Y., Dong, K., Li, Y., Ding, J., & Han, X. (2024, April). Detecting Anomalies in LiDAR Point Clouds. At WCX SAE World Congress Experience. SAE Technical Paper.

[7] Ding, J. (2025). Exploration of Process Improvement in Automotive Manufacturing Based on Intelligent Production. International Journal of Engineering Advances, 2(2), 17-23.

[8] Wu, Y. (2026). Federated Learning-based Algorithm Design for Privacy Preservation in Cross-domain Data Sharing. Engineering Advances, 6(1).

[9] Sun, J. (2025). Research on Business Data-driven Risk Prediction Methods Based on Machine Learning. Advances in Computer and Communication, 6(4).

[10] Yanchun Wang. (2025) Research on Improving ERP System Efficiency with AI in Cross-border Supply Chain Environments. Advances in Computer and Communication, 6(5), 268-273.

[11] Zhou, Y. (2026). Energy efficiency and sustainability strategies for data centers. European Journal of Engineering and Technologies, 2(1), 46-53.

[12] Lu, Z. (2025). Design and Practice of AI Intelligent Mentor System for DevOps Education. European Journal of Education Science, 1(3), 25-31.

[13] Wu Y. Software Engineering Practice of Microservice Architecture in Full Stack Development: From Architecture Design to Performance Optimization[J]. 2025.

[14] Liu, X., & Yang, D. (2025, March). LLM Data Strategy: Improving Data Availability and Efficiency. In Doctoral Symposium on Computational Intelligence (pp. 425-437). Singapore: Springer Nature Singapore.

[15] Zhang, Q. (2025, October). Application of Reinforcement Learning in Dynamic Advertising Content Generation. In 2025 2nd International Conference on Software, Systems and Information Technology (SSITCON) (pp. 1-5). IEEE.

[16] Zhang, Q. (2026). Security Improvement and Application of Identity and Access Management in Saas Platform.

[17] Hou, Y. (2026). Research on Server Performance Stability Assurance Mechanisms during Cross-Generation Computing Platform Upgrades.

[18] Wu, Y. (2025, October). Multi-Level Belief Rule Base Modeling Architecture and Intelligent Optimization Technology for Decision Support Systems. In 2025 2nd International Conference on Software, Systems and Information Technology (SSITCON) (pp. 1-8). IEEE.

[19] Chen, M. (2026). Research on Privacy-Preserving AI Model Training and Validation Methods Based on Federated Learning.

[20] Yiting Hong. Differentially Private High-Dimensional Business Data Publishing and Analysis Algorithm. International Journal of Business Management and Economics and Trade (2026), Vol. 7, Issue 1: 28-35.

[21] Xu, D. (2026). Analysis of the impact of video infrastructure optimization on large-scale content quality improvement.

[22] Chen M. Research on Automated Risk Detection Methods in Machine Learning Integrating Privacy Computing[J]. 2025.

[23] Pan, H. (2025, March). Research on Efficient Computing Model of Hartree Fock and Density Functional Theory Based on GPU Acceleration. In Doctoral Symposium on Computational Intelligence (pp. 485-496). Singapore: Springer Nature Singapore.

[24] Wu, W. (2025, June). Construction and optimization of intelligent gateway software management platform based on jenkins cluster management under cloud edge integration architecture in industrial internet of things. In International Conference on 6G Communications Networking and Signal Processing (pp. 633-645). Singapore: Springer Nature Singapore.

[25] Wu Y. Software Engineering Practice of Microservice Architecture in Full Stack Development: From Architecture Design to Performance Optimization[J]. 2025.

[26] Wu Y. Optimization of Generative AI Intelligent Interaction System Based on Adversarial Attack Defense and Content Controllable Generation[J]. 2025.

[27] Sun J. Quantile Regression Study on the Impact of Investor Sentiment on Financial Credit from the Perspective of Behavioral Finance[J]. 2025.

[28] Wang Y. Application of Data Completion and Full Lifecycle Cost Optimization Integrating Artificial Intelligence in Supply Chain[J]. 2025.

[29] Yu, X. (2026). Strategy Models and Practical Research of Growth Marketing under the Background of Digital Transformation.

[30] Hou, Y. (2026). Research on BIOS and BMC Compatibility Optimization Methods for Cross-Generation Servers in Production Environments.