Universal Library of Engineering Technology Research Article

ISSN: 3064-996X | Volume 2, Issue 3

Open Access | PP: 119-124

DOI: https://doi.org/10.70315/uloap.ulete.2025.0203022

Architectural Approaches to Real-Time AI-Based Risk Monitoring Systems for Critical Infrastructure

Ilia Serebriakov

Engineer, Student, The City University of New York, Engineering Science

Abstract

This article analyzes architectural approaches to building real-time risk monitoring systems in the context of the development of intelligent technologies in critical infrastructure. The study is conducted in the format of a systematic review and analytical synthesis of academic publications focusing on the application of artificial intelligence in monitoring systems, the specifics of stream data processing, distributed computing, and mechanisms of explainability and control. The primary focus is on the relationship between the characteristics of the digital environment, the logic of architectural organization of information processing, and the formation of risk signals. Key factors determining monitoring effectiveness are examined, including data quality, algorithm performance, distribution of computation, and control mechanisms. It is established that the influence of intelligent technologies is indirect and is realized through changes in the structure of information formation across system levels. It is shown that risk monitoring ceases to be a procedure of deviation detection and acquires a systemic character, being formed within the architecturally coordinated interaction of data, algorithms, and governance. An original model is developed, reflecting the sequential interaction of data streams, analytical models, mechanisms of explainability, and control in the process of risk signal formation. The results obtained make it possible to consider risk monitoring as an architecturally determined process that defines the stability and reliability of system functioning. The article will be useful for researchers in intelligent systems, specialists in the design of distributed architectures, and practitioners involved in the development and implementation of monitoring systems in critical infrastructure.

Keywords: *Risk Monitoring, Critical Infrastructure, Artificial Intelligence, System Architecture, Stream Data, Distributed Computing, Explainability.*

INTRODUCTION

Critical infrastructure comprises interconnected technical, informational, and organizational systems, including energy networks, urban complexes, transport hubs, and distributed production facilities [3]. As digital solutions expand, both the volume of data and the number of interconnections between processes and management increase, causing even local failures to propagate rapidly across the system. Risk is not formed as an isolated failure; it emerges from the interaction of data streams, processing stages, computational procedures, and operator actions. Traditional monitoring systems are oriented toward capturing system states or individual deviations, and in such an environment, they are no longer sufficient.

A transition is taking place toward real-time data processing and decision support based on multi-level systems that integrate data collection, processing, and interpretation [7]. New solutions are often focused on individual analytical methods. However, issues related to result interpretability, distribution of computation, system resilience, and integration with management remain insufficiently addressed. A holistic architecture is lacking. The system does not integrate data processing, analysis, explanation, and control into a unified framework. This fragmentation reduces the practical value of solutions and necessitates the development of a unified architectural approach to risk monitoring in critical infrastructure.

The aim of the study is to develop and theoretically substantiate an original architectural model for real-time AI-based risk monitoring systems, focusing on the integration of data processing, analytical models, explainability, and governance mechanisms. To achieve this aim, the following objectives are set:

– to analyze existing architectural approaches to real-time risk monitoring systems;

Citation: Ilia Serebriakov, "Architectural Approaches to Real-Time AI-Based Risk Monitoring Systems for Critical Infrastructure", Universal Library of Engineering Technology, 2025; 2(3): 119-124. DOI: https://doi.org/10.70315/uloap.ulete.2025.0203022.

– to identify the main system levels, from data collection to warning generation;

– to assess the role of distributed infrastructure, data processing, and models in ensuring the resilience and continuity of monitoring.

The research hypothesis is that the effectiveness of risk monitoring is determined not by the quality of individual models, but by the coherence of system levels. Reliable operation is achieved when data collection, processing, analysis, explanation, and control are integrated into a unified loop. In the absence of such coherence, even highly accurate models lose their practical value due to weak interpretability of results and errors arising at earlier stages of processing.

The scientific novelty lies in considering risk monitoring systems as architecturally coordinated loops operating under conditions of incomplete, asynchronous, and distorted data, which is largely underrepresented in existing research. In contrast to the dominant approach focused on the accuracy of individual models, it is shown that such an orientation reduces system reliability due to the disconnection between stages of processing, analysis, and control. The study develops a formalized architectural scheme describing the interrelation of system levels as a single continuous loop. Explainability is treated as an embedded system layer rather than an external tool. The proposed approach makes it possible to evaluate monitoring effectiveness through architectural coherence and resilience to data distortions, thereby establishing a different principle for the design and analysis of such systems.

MATERIALS AND METHODS

The study follows a qualitative conceptual research design based on a systematic literature review and analytical synthesis. The study is based on methods of theoretical analysis of academic publications, conceptual systematization of architectural approaches, and comparative analysis of factors determining the effectiveness of real-time risk monitoring systems. The primary focus is on identifying relationships between the characteristics of data streams, the organization of the computational environment, and the structure of architectural solutions that ensure the generation of interpretable risk signals.

The study was conducted in the format of a systematic review of open-access academic publications from 2022–2025, presented in international peer-reviewed journals and academic databases. The literature search was carried out in Google Scholar, ScienceDirect, SpringerLink, and MDPI using combinations of keywords: “artificial intelligence risk monitoring,” “intelligent energy systems,” “real-time anomaly detection,” “distributed computing systems,” “stream data processing,” and “decision governance and explainability,” applying logical operators AND/OR. The sample included English-language publications containing theoretical or applied analyses of architectural solutions, data processing, and intelligent analytics in monitoring systems. Studies focused exclusively on individual algorithms without considering system architecture were excluded.

At the identification stage, 48 publications were selected. After removing duplicates and screening titles and abstracts, irrelevant sources were excluded. Full-text analysis resulted in a final sample of 14 publications corresponding to the research objectives.

The analytical procedure included sequential stages: source identification, duplicate removal, thematic selection, full-text analysis, and conceptual classification of findings. During the analysis, the following categories were identified: data stream structure, levels of processing and analysis, characteristics of distributed infrastructure, mechanisms of result interpretation, and system governance elements. The comparison of results was conducted through the analysis of the influence of these factors on architectural coherence and the system’s ability to generate stable risk signals.

The limitation of the sample is related to the fact that a significant portion of publications in the monitoring domain focuses on improving the accuracy of individual methods, while architectural aspects are addressed only fragmentarily. The analyzed studies cover issues of data processing, intelligent analytics, distributed computing, and governance, but rarely integrate them into a unified model.

The results obtained were used to systematize the architectural levels of risk monitoring systems and to develop an original model reflecting the coordinated interaction of data, processing, analysis, interpretation, and governance under real-time conditions.

RESULTS

The analysis reveals that risk monitoring systems exhibit a stable multi-layer architectural structure, where system performance is determined by the interaction between data, computation, and governance rather than by individual models. Within the framework of the study, the architecture of real-time risk monitoring systems is considered as a distributed environment in which data, computation, and control are linked by rigid dependencies. These dependencies determine system behavior under load. As the volume of incoming data increases, scaling does not occur linearly. Instead, functions are redistributed across system layers. This effect is observed in intelligent energy networks, transport systems, and urban infrastructure [1, 3]. Under such conditions, architecture ceases to play a supporting role and begins to constrain accuracy, latency, and system resilience.

The analysis shows that risk formation is not limited to the model level. A significant proportion of errors arises at the stages of data transmission, preprocessing, and result interpretation [9]. Errors may occur before the algorithm is executed or after the output is generated. This shifts the

analytical focus from the algorithm to the architectural organization. A similar pattern is observed in distributed signal processing systems, where latency and packet loss directly affect anomaly detection [11]. Data streams in such systems have a complex structure, including time series and events [9]. As a result, the analysis is conducted along two dimensions: first, the identification of a stable multilayer structure; second, the examination of its implementation at the infrastructure and data levels.

A comparison of architectures across different systems reveals a recurring structural pattern. Although formal descriptions may vary, the underlying organizational logic remains consistent. Four layers can be identified: governance, technology, design, and operations. Each layer imposes constraints on adjacent ones and defines permissible modes of system functioning. Table 1 presents the distribution of parameters across architectural layers and their impact on risk monitoring.

Table 1. Architectural distribution of parameters in real-time AI-based risk monitoring systems (Compiled by the author based on source: [3])

Architectural domain	Number of parameters	Significance for risk monitoring
AI Behaviour and Governance	1	Defines accountability, explainability and decision constraints
Technology	4	Determines data acquisition, transmission and protection mechanisms
Design and Development	3	Shapes model behaviour, data representation and transparency
Operations	7	Executes forecasting, anomaly detection and system response

The distribution of parameters reveals pronounced asymmetry. The operational layer concentrates the core functions of detection and response. This concentration is explained by the need to process continuous data streams without significant latency. However, the governance layer is minimally represented. Governance defines requirements for explainability and control but does not determine system behavior in real-time operation.

A contradiction emerges. Rapid detection of deviations is not accompanied by equally rapid explanation of their causes. Increased processing speed further reduces interpretability. A direct constraint between latency and explainability is formed. In critical infrastructure systems, this conflict becomes systemic. Achieving both minimal latency and full interpretability simultaneously requires a revision of architectural logic.

The technological layer captures the distribution of computation. Centralized processing increases load and leads to higher latency. A distributed scheme reduces the burden on the central node but complicates coordination [14]. This gives rise to a second contradiction: edge versus accuracy. Local processing reduces latency but limits analytical depth, whereas centralized processing increases accuracy but prolongs response time.

The design layer connects these constraints. The choice of models and data representation methods determines computational requirements. The use of generative models increases sensitivity to anomalies but also raises system load [8]. As a result, the architecture must either redistribute resources or reduce the frequency of analysis. In this configuration, the multilayer structure reflects not merely a set of components but a system of constraints. Each layer imposes its own requirements, and their simultaneous fulfillment necessitates trade-offs.

The practical implementation of a multilayer structure requires a distributed infrastructure. As the number of sensor nodes increases, a centralized scheme can no longer provide acceptable latency. Communication channel capacity becomes a limiting factor. Table 2 presents the main hardware and communication parameters.

Table 2. Hardware and communication parameters of distributed real-time AI architecture (Compiled by the author based on sources: [3, 14])

Parameter	Value	Architectural role
Inter-node link bandwidth	High (Gbit-class)	Parallel data transfer between nodes
Number of parallel links	Multiple	Throughput scaling between distributed nodes
High-speed serial communication	Multi-Gbit/s	High-throughput data exchange in distributed topology
FPGA processing frequency	Hundreds of MHz	Real-time synchronization across compute nodes

The specified parameters determine performance limits. Exceeding throughput capacity leads to queue accumulation and increased latency. Even with high processing rates at individual nodes, synchronization is disrupted. Under such conditions, the redistribution of computation to the edge becomes necessary.

Data streams are characterized by heterogeneity, including images, signals, events, and metadata. The volume of information grows faster than computational capabilities. As a result, the architecture shifts toward preliminary processing at the edge. Filtering and initial validation are performed first, after which the data are transmitted to higher levels. Table 3 presents the parameters of the training and testing datasets.

Table 3. Training and testing datasets for AI-based infrastructure monitoring models (Compiled by the author based on source: [1])

Detection object	Training dataset, images	Testing dataset, images
Damaged road signs	2917	731
Dumped rubbish	7808	1956
Bus shelters	1004	252

The distribution of data reveals a pronounced imbalance. Frequent classes are represented far more extensively than rare ones. Such a distribution affects model behavior: the accuracy of recognizing common events increases, while sensitivity to rare events decreases. In risk monitoring tasks, this effect is critical, as rare events are often associated with the greatest threats. Compensation is achieved through cascading processing and repeated validation at different levels [12].

A hybrid architecture, with computation distributed between edge and cloud levels, introduces an additional constraint. Local nodes provide rapid response, while control and auditing are concentrated at higher levels. This creates a gap between decision-making and its verification. A conflict emerges between speed and governance, and the system begins to operate in a mode of delayed control.

Infrastructure and data jointly determine the architectural configuration. Changes in the volume or structure of data streams lead to a redistribution of functions, shifts in decision points, and adjustments in the balance between speed, accuracy, and control.

DISCUSSION

In modern infrastructure monitoring systems, the key factor is no longer the choice of an individual model, but the organization of the entire data processing architecture. Practice demonstrates a shift from isolated algorithms to interconnected computational loops. The model ceases to be the central element. The governing role is assumed by the structure of the data flow and the way it traverses the system.

The limitations of the single-model approach become evident as system load increases and the environment becomes more complex. A single algorithm cannot maintain stability under changing data distributions [13]. Even advanced models lose accuracy when data streams become irregular. As a result, the architecture begins to reconfigure. A processing chain is formed, where each stage performs a distinct function. This logic does not enhance the model itself but redistributes responsibility across system components.

Multilevel integration emerges as a consequence of this transition. However, increasing the number of layers does not resolve the problem but shifts it. Control is strengthened, but latency increases. Reducing the number of layers produces the opposite effect—speed improves, but the transparency of decisions decreases. A persistent contradiction arises that cannot be eliminated at the architectural level. Any configuration merely reflects a current balance.

Distributed architectures further intensify this tension. Asynchronous processing reduces data transmission latency [2]. Data streams do not block one another, and the system responds more quickly. However, maintaining consistency becomes more complex. At high event frequencies, inconsistencies accumulate. Errors do not always manifest immediately; they may propagate through multiple stages of processing. Additional pressure is introduced by AIoT and mobile sensing systems [1]. Data streams become uneven, with transmission frequency depending on environmental and device conditions. A single system may simultaneously involve thousands of sensors generating data at different rates. Under such conditions, centralized architectures cease to operate reliably.

A hybrid architecture, distributing computation between edge and cloud levels, amplifies internal system contradictions. Edge processing reduces latency and enables rapid response, while cloud processing provides more accurate analysis based on complete datasets. Their simultaneous use creates a direct conflict: accelerating processing limits analytical depth, whereas improving accuracy increases response time. Figure 1 presents the integrated architecture of the risk monitoring system developed in this study.

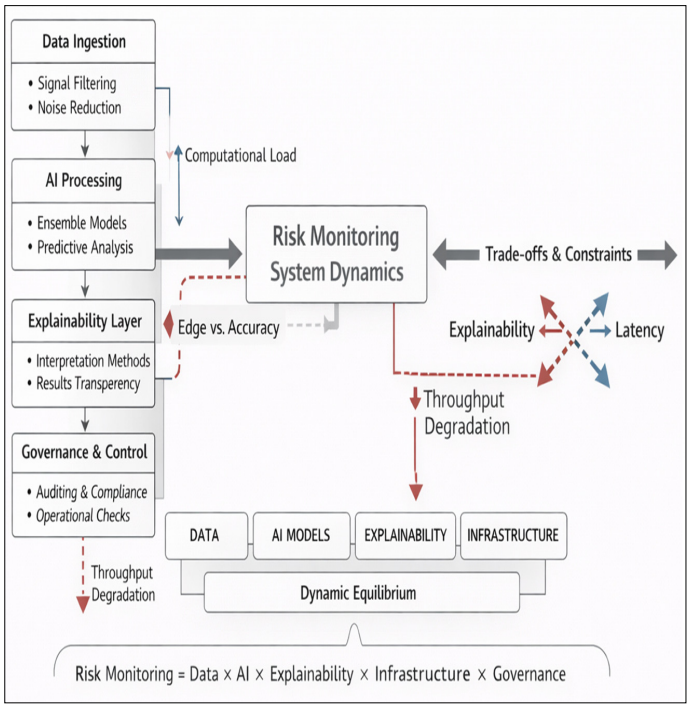

Figure 1. Integrated architecture of a real-time AI-based risk monitoring system (Author's development)

The proposed architecture is not reduced to a formal division into layers. It describes the system as a connected configuration of constraints arising in real-time data stream processing. The flow received from sensor nodes inherently contains distortions. Noise, missing data, and unstable transmission frequency create initial deviations. At early stages, filtering and signal reconstruction are performed. If a deviation passes this stage, it is amplified as it propagates further.

Subsequent processing is based on a combination of models. Ensembles are used, including transformer-based and generative approaches. Increasing the number of models improves result robustness but simultaneously raises computational load. Gains in accuracy directly increase processing time. Thus, a hard constraint emerges that cannot be eliminated by increasing model complexity.

Attempts to enhance the transparency of decisions alter system behavior. Interpretability methods increase trust in results; however, each explanation procedure adds latency. In real-time environments, even small increases in processing time become critical. Transparency begins to constrain speed rather than enhance it. Control mechanisms introduce an additional layer of constraints. Auditing and compliance requirements add verification stages throughout processing. These checks improve reliability but reduce system throughput. Under high load, control itself becomes a limiting factor for scalability.

The final stage connects the system with the operator. Signals and feedback form a closed loop. At this stage, the cumulative effect of all preceding constraints becomes evident. Latency introduced at earlier stages is not compensated, and errors are not isolated. System response is determined not by a single component but by the entire architecture. The proposed model describes architecture as a function of interdependent factors:

Risk Monitoring = f (Data × AI × Explainability × Infrastructure × Governance)

Each element influences system behavior. Strengthening one component inevitably constrains others. Increasing speed reduces explainability. Enhancing control increases latency. Expanding infrastructure improves scalability but complicates the coordination of data flows.

A fixed optimal configuration in such systems is unattainable. The architecture is formed as a dynamic equilibrium that continuously shifts under the influence of load, data structure, and governance requirements. System behavior is determined not by the choice of individual technologies, but by the ability to maintain this balance under changing conditions.

CONCLUSION

The results obtained make it possible to consider risk monitoring in critical infrastructure not as a set of individual algorithms or processing stages, but as a process formed within an architecturally coordinated system operating under conditions of continuous data flow. Its content is determined not only by the characteristics of models but also by the structure of interactions between the levels of data collection, computation, interpretation, and control. Under these conditions, the formation of risk signals occurs not at the level of a single analytical module, but within the space of coordination between data streams, computational procedures, and control mechanisms.

The coherence of architectural elements becomes decisive. Even with high accuracy of individual components, such as algorithms or data sources, the final result remains unstable if there is a disconnect between the stages of data transmission, processing, and interpretation. When coherence between system levels is ensured, an integrated logic of operation emerges, within which risk signals become more stable and interpretable. In this configuration, architecture ceases to be a supporting environment and begins to perform the function of integrating and balancing the constraints that determine system behavior.

The practical significance of the study lies in the need to reconsider approaches to designing risk monitoring systems. The stability and reliability of results depend on the ability to establish coordinated interaction between data streams, distributed computational infrastructure, analytical models, and mechanisms of explainability and control. Technological solutions remain important; however, their effectiveness is limited in the absence of architectural integrity and mechanisms ensuring coordination across system levels.

The research hypothesis is confirmed, as it is demonstrated that the effectiveness of risk monitoring is determined not by the quality of individual models, but by the degree of coherence among architectural levels that integrate data, algorithms, interpretation, infrastructure, and governance into a unified continuous loop. As a result, risk monitoring transforms from a set of isolated procedures into a dynamic architectural system whose functioning is determined by the balance of interrelated constraints. The results can be applied in the design of next-generation monitoring systems for energy, transportation, and urban infrastructure, where real-time decision-making requires a balance between speed, accuracy, and interpretability.

REFERENCES

1. Forkan, A. R. M., Kang, Y. B., Marti, F., et al. (2024). AIoT-CitySense: AI and IoT-driven city-scale sensing for roadside infrastructure maintenance. Data Science and Engineering, 9, 26–40. https://doi.org/10.1007/s41019-023-00236-5
2. Fragkoulis, M., et al. (2024). Stream processing systems: state-of-the-art and open challenges. VLDB Journal. https://doi.org/10.1007/s00778-023-00819-8
3. Alsaigh, R., Mehmood, R., & Katib, I. (2023). AI explainability and governance in smart energy systems:

A review. Frontiers in Energy Research, 11. https://doi.org/10.3389/fenrg.2023.1071291

4. Choi, A., Lee, K., Hyun, H., et al. (2024). A novel deep learning algorithm for real-time prediction of clinical deterioration in the emergency department for a multimodal clinical decision support system. Scientific Reports, 14, 30116. https://doi.org/10.1038/s41598-024-80268-7
5. Duan, J. (2024). Deep learning anomaly detection in AI-powered intelligent power distribution systems. Frontiers in Energy Research, 12. https://doi.org/10.3389/fenrg.2024.1364456
6. Belay, M. A., Blakseth, S. S., Rasheed, A., & Salvo Rossi, P. (2023). Unsupervised anomaly detection for IoT-based multivariate time series: Existing solutions, performance analysis and future directions. Sensors, 23(5), 2844. https://doi.org/10.3390/s23052844
7. Jian, M.-S., & Pan, C.-J. (2022). Blockchained industry information handoff based on Internet of Things devices with intelligent customized object recognition. Sensors, 22(6), 2312. https://doi.org/10.3390/s22062312
8. Choi, A., & Kim, H. (2024). Anomaly detection in industrial systems. Applied System Innovation. https://doi.org/10.3390/asi7020018
9. Hao, W., Yang, T., & Yang, Q. (2023). Hybrid statistical-machine learning for real-time anomaly detection in industrial cyber-physical systems. IEEE Transactions on Automation Science and Engineering, 20, 32-46. https://doi.org/10.1109/TASE.2021.3088805
10. Nemade, B., Maharana, K. K., Kulkarni, V., et al. (2024). Revolutionizing smart grid security: A holistic cyber defence strategy. Frontiers in Artificial Intelligence, 7. https://doi.org/10.3389/frai.2024.1476422
11. Hoang, N. X., et al. (2022). Explainable Anomaly Detection for Industrial Control System Cybersecurity. IFAC-PapersOnLine, 55(36), 223–228. https://doi.org/10.1016/j.ifacol.2022.12.038
12. Andriulo, L., et al. (2024). Edge vs cloud computing in IoT systems. Informatics. https://doi.org/10.3390/informatics11040071
13. Isah, H., et al. (2019). A scalable distributed framework for stream processing. IEEE Access. https://doi.org/10.1109/ACCESS.2019.2946884
14. Sodiya, A. S., et al. (2024). Edge computing in IoT: Architectures and challenges. IJSRA. https://doi.org/10.30574/ijsra.2024.11.1.0287