

Article

Not peer-reviewed version

A Low-Altitude Urban Air Mobility Operations Platform with Mission Lifecycle Assetization

[Yushu Liu](#)^{*}, Longbiao Wang, Chenglin Du, Haixiao Zhai

Posted Date: 30 December 2025

doi: 10.20944/preprints202512.2648.v1

Keywords: urban air mobility; low-altitude operations; drone operations platform; mission lifecycle management; UAM systems architecture; operational platforms



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

A Low-Altitude Urban Air Mobility Operations Platform with Mission Lifecycle Assetization

Yushu Liu ^{1,*}, Longbiao Wang ¹, Chenglin Du ² and Haixiao Zhai ³

¹ Tianjin Key Laboratory of Cognitive Computing and Application, College of Intelligence and Computing, Tianjin University, China

² Executive Deputy Director and Secretary-General of the Information Technology Sub-committee of the Central Committee of the China Democratic National Construction Association, China

³ Officer, Police Aviation Corps Hunan, China

* Correspondence: yushuliu@outlook.com

Abstract

Urban Air Mobility (UAM) and low-altitude drone operations are emerging as a critical component of next-generation urban transportation and logistics systems. As mission volumes increase, operators face growing challenges in coordinating large-scale low-altitude missions, managing heterogeneous operational states, and closing the loop between mission execution and platform-level resource utilization. Existing UAM platforms primarily focus on flight scheduling and monitoring, while lacking systematic mechanisms to model mission lifecycles and their operational value within an integrated platform architecture. This paper presents SkyNetUAM, a low-altitude UAM operations platform that introduces a structured mission lifecycle model to bridge mission planning, execution, and post-mission settlement within a unified system. We propose a hierarchical operational asset model covering individual missions, service packages, and air-corridor time slots, enabling fine-grained tracking of mission states and operational performance. The platform architecture integrates real-time mission scheduling, operational monitoring, and lifecycle state management, allowing mission-level events to drive platform-wide coordination and resource allocation. As an operational extension, the system incorporates a lightweight on-chain persistence mechanism to record mission states and support automated settlement workflows without altering the core operational logic. A prototype implementation demonstrates the end-to-end workflow from mission creation to completion across simulated low-altitude scenarios, and a reproducible 100k-missions/day experiment quantifies approval rate, delay behavior, and latency distributions under congestion and regulatory constraints.

Keywords: urban air mobility; low-altitude operations; drone operations platform; mission lifecycle management; UAM systems architecture; operational platforms

1. Introduction

1.1. Low-Altitude Urban Air Mobility and Drone Operations

Urban Air Mobility (UAM) and low-altitude drone operations are rapidly emerging as a new layer of urban infrastructure, enabling applications ranging from logistics and inspection to passenger transport and emergency response. Advances in electric vertical take-off and landing (eVTOL) vehicles, autonomous flight control, and communication technologies have accelerated the deployment of dense aerial operations in urban airspace. As cities begin to treat airspace below approximately 1,000 meters as an economic and operational domain, the concept of a "low-altitude economy" has gained increasing attention from both industry and policymakers.

Unlike traditional aviation, low-altitude UAM operations are characterized by high mission frequency, short flight durations, heterogeneous vehicle capabilities, and strong coupling with urban infrastructure. Missions are dynamically generated, frequently updated, and executed under rapidly changing environmental and regulatory conditions. These characteristics impose significant challenges

on system-level coordination, including mission scheduling, airspace usage, fleet management, and real-time monitoring. As a result, scalable digital platforms are widely recognized as a prerequisite for enabling reliable and efficient low-altitude drone and UAM operations.

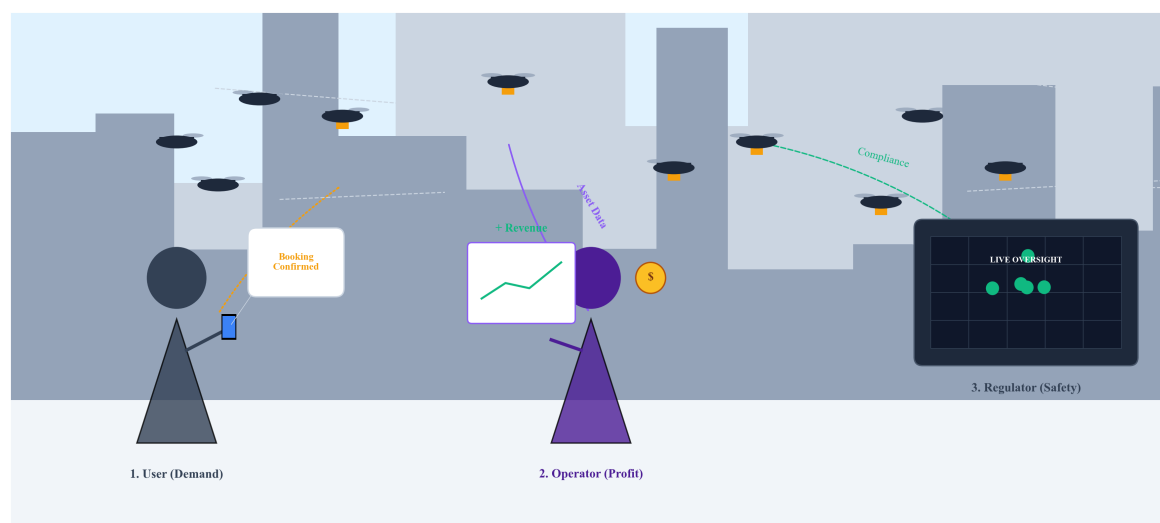


Figure 1. Operational scenario of the SkyNetUAM platform. (1) A user books a low-altitude service via a mobile terminal. (2) **An operator receives asset data and monitors revenue streams, realizing the economic value of the flight.** (3) Regulators maintain real-time oversight of airspace and flight telemetry through a centralized monitoring interface. This end-to-end workflow integrates user demand, commercial operation, and regulatory compliance.

1.2. Limitations of Existing UAM Operational Platforms

Existing UAM and drone operation platforms primarily focus on flight planning, trajectory management, and real-time monitoring. Industrial platforms and research prototypes typically provide functionalities such as mission request intake, route planning, fleet status visualization, and basic operational alerts. While these capabilities are essential, they often treat missions as transient execution tasks rather than persistent operational entities with explicit lifecycle states.

In practice, each UAM mission progresses through multiple stages, including booking, scheduling, execution, completion, cancellation, or failure. However, most current platforms lack a unified mechanism to represent these mission lifecycle states in a structured and machine-readable manner. Mission-related data are frequently fragmented across different system components, making it difficult to consistently track operational status, evaluate performance, or coordinate downstream processes such as settlement, reporting, and post-mission analysis.

This limitation becomes increasingly critical as mission density grows. Without lifecycle-aware modeling, platforms struggle to manage high levels of concurrency, recover from disruptions, or provide reliable system-level guarantees. Consequently, there is a growing need for UAM platforms that explicitly model missions as first-class operational entities and integrate lifecycle management into the core system architecture.

1.3. Mission Lifecycle Modeling and Platform-Level Integration

To address these challenges, recent research has emphasized the importance of platform-centric designs that integrate mission execution with system-wide coordination mechanisms. From this perspective, a UAM platform should not only orchestrate flights but also manage mission states, transitions, and dependencies across the entire operational workflow. Treating missions as structured operational assets enables platforms to reason about availability, reliability, and performance at a higher level of abstraction.

In this work, we adopt a mission-centric approach in which individual missions, service packages, and air-corridor time slots are modeled as lifecycle-aware operational entities. Each entity is associated

with well-defined states and transitions driven by operational events such as mission start, completion, delay, or failure. This abstraction allows mission execution data to be consistently propagated across scheduling, monitoring, and post-mission processes within the same platform.

As an operational extension, the platform supports optional persistence of mission lifecycle states using on-chain mechanisms. This design choice enables automated state verification and settlement workflows while preserving the primacy of the operational logic within the UAM platform. Importantly, the introduction of on-chain components does not alter flight control, scheduling algorithms, or safety-critical functions; instead, it serves as a supporting layer for state synchronization and system integration.

1.4. Contributions and Paper Organization

Building on the above motivation, this paper presents **SkyNetUAM**, a lifecycle-aware operations platform for low-altitude drone and UAM systems. The main contributions of this work are as follows:

- We design a unified platform architecture that integrates mission scheduling, execution monitoring, and lifecycle state management for large-scale low-altitude UAM operations.
- We propose a mission-centric operational modeling framework that formalizes missions, service packages, and air-corridor resources as structured platform entities.
- We introduce an extensible mechanism for mission state persistence and automated settlement as an operational extension to the core UAM platform.
- We implement a working prototype and evaluate its performance under high mission concurrency, demonstrating improved scalability, reduced operational latency, and enhanced system stability.

The remainder of this paper is organized as follows. Section 2 reviews related work on UAM platforms, low-altitude operations, and supporting digital infrastructures. Section 3 describes the system architecture and platform design. Section 4 presents the operational modeling framework and lifecycle mechanisms. Section 5 details the prototype implementation, followed by experimental evaluation in Section 6. Section 7 discusses limitations and future research directions, and Section 8 concludes the paper.

2. Related Work

This section reviews prior research relevant to low-altitude drone and Urban Air Mobility (UAM) operations from four perspectives: (i) UAM and low-altitude operational systems, (ii) digital platforms for UAM and Advanced Air Mobility (AAM), (iii) blockchain and distributed ledgers in drone and mobility systems, and (iv) operational asset modeling and lifecycle management. The review concludes with a positioning of this work within the existing literature.

2.1. Urban Air Mobility and Low-Altitude Drone Operations

Urban Air Mobility has been extensively studied as an emerging transportation paradigm for dense metropolitan environments. Early surveys and reviews focus on vehicle concepts, airspace integration, demand modeling, and regulatory challenges associated with deploying autonomous aerial vehicles in urban settings. These studies emphasize that UAM operations differ fundamentally from traditional aviation due to higher mission frequency, shorter flight durations, and tighter coupling with urban infrastructure.

Recent research has extended this perspective by framing low-altitude airspace as a distinct operational and economic domain. The notion of a “low-altitude economy” highlights the increasing use of drones and eVTOLs for logistics, inspection, emergency response, and passenger transport within constrained urban airspace. In this context, the primary challenges shift from individual vehicle performance to system-level coordination, including mission scheduling, airspace capacity allocation, and operational resilience under high mission density.

While this body of work establishes the technological and regulatory foundations for UAM, most studies focus on flight-level considerations such as routing, collision avoidance, and airspace

management. Comparatively less attention has been paid to how missions are represented, tracked, and managed as operational entities within large-scale UAM platforms.

2.2. Digital Platforms for UAM and Advanced Air Mobility

Digital platforms are increasingly recognized as the operational backbone of UAM and Advanced Air Mobility ecosystems. Both industrial initiatives and academic prototypes propose integrated software platforms that combine mission planning, fleet management, monitoring, and user interaction within a unified system. These platforms aim to provide real-time situational awareness and decision support for operators managing heterogeneous aerial fleets.

Industrial systems such as proprietary UAM operation centers and vertiport management platforms emphasize end-to-end visibility of vehicle status, mission progress, and infrastructure utilization. Similarly, research platforms explore modular architectures that decouple user interfaces from backend services, enabling scalability and extensibility. Common platform functionalities include mission request intake, route planning, telemetry visualization, and basic performance monitoring.

Despite these advances, existing platforms typically treat missions as transient execution tasks. Mission data are often tightly coupled to specific services or logs, limiting their reuse across scheduling, monitoring, and post-operation processes. As mission volumes increase, this lack of explicit mission lifecycle modeling becomes a bottleneck for scalability, coordination, and system-level optimization.

2.3. Blockchain and Distributed Ledgers in Drone and Mobility Systems

Blockchain and distributed ledger technologies have been explored as enabling infrastructures for trust, transparency, and automation in mobility and cyber-physical systems. In the context of smart cities and Mobility-as-a-Service, blockchain has been applied to integrated ticketing, access control, and decentralized service coordination, demonstrating its potential to support multi-stakeholder environments.

Within the UAM and drone domain, prior work has investigated the use of distributed ledgers for sharing operational intent, synchronizing flight plans, and maintaining tamper-evident logs among operators and regulators. These studies show that blockchain can facilitate secure data exchange and coordination without relying on a single centralized authority. Other research explores the tokenization of non-flight resources, such as onboard computing capacity or communication bandwidth, treating them as tradable infrastructure assets.

However, most blockchain-related work in UAM focuses on data sharing or resource markets rather than core operational workflows. The integration between mission execution, lifecycle state management, and platform-level coordination remains largely unexplored. As a result, blockchain is often introduced as an external service rather than being embedded as a supporting component within UAM operational platforms.

2.4. Operational Asset Modeling and Mission Lifecycle Management

Modeling operational entities and their lifecycles is a well-established concept in complex system engineering and digital platform design. In transportation and logistics systems, lifecycle-aware modeling enables consistent tracking of tasks, resources, and performance across planning, execution, and evaluation stages. Treating tasks as first-class entities with explicit states supports scalability and fault recovery in high-throughput environments.

In the context of low-altitude drone and UAM operations, missions naturally exhibit asset-like characteristics: they are uniquely identifiable, temporally bounded, and associated with measurable performance outcomes. Recent studies have begun to explore structured representations of missions and services to support analytics and optimization. Nevertheless, these efforts often remain disconnected from platform-level architectures and do not provide an end-to-end mechanism linking mission lifecycle states to operational coordination.

This work builds upon these insights by embedding mission lifecycle modeling directly into a UAM operations platform. By treating missions, service packages, and airspace resources as lifecycle-

aware operational entities, the proposed approach enables consistent state management across scheduling, execution, and settlement processes. Compared with prior work, the emphasis is not on financial abstraction but on improving system-level coordination and operational reliability in large-scale low-altitude environments.

2.5. Summary and Positioning of This Work

In summary, existing research on UAM and low-altitude drone operations provides strong foundations in vehicle technology, airspace management, and digital platform design. Blockchain-based approaches contribute valuable tools for secure data sharing and automation but are typically applied as external services. There remains a gap in end-to-end platform designs that explicitly integrate mission lifecycle modeling with operational coordination mechanisms.

The proposed SkyNetUAM platform addresses this gap by introducing a lifecycle-aware operational architecture for low-altitude drone and UAM systems. By embedding mission-centric modeling and optional state persistence mechanisms into the platform itself, this work complements existing UAM research and aligns with the growing emphasis on scalable, system-oriented solutions for future low-altitude operations.

Table 1. Representative related work and scope comparison (conceptual).

Category	Representative Work	Primary Focus	Platform?	RWA Target
UAM Reviews & ConOps	Garrow et al. [1]	UAM vehicle types, demand modeling, and regulations	No	N/A
	EHang Whitepaper [2]	Autonomous aerial vehicle systems and vertiport infrastructure	No	N/A
Industrial Platforms	Volocopter (VoloIQ) [3]	Digital backbone for flight planning and fleet management	Yes	None (Ops only)
	EHang Command	Real-time monitoring and cluster management	Yes	None (Ops only)
Blockchain & Mobility	Miron et al. [4]	Blockchain integration for MaaS platforms in smart cities	Partial	Access Rights
	Freeman et al. [5]	Operational Intent (OI) exchange and deconfliction	No	Data (Logs)
Resource RWA	Luo et al. [6]	RWA-assisted architecture for low-altitude computility networks	No	Computility
This Work	SkyNetUAM	Lifecycle-aware operations platform with optional state persistence	Yes	Missions, Slots, Service Packages

3. System Architecture of the SkyNetUAM Platform

This section presents the system architecture of the proposed SkyNetUAM platform for low-altitude drone and Urban Air Mobility (UAM) operations. The design follows a mission-centric and platform-oriented approach, in which mission workflows, operational state management, and system coordination are treated as first-class concerns. The architecture integrates mission scheduling, execution monitoring, and lifecycle state propagation within a unified operational platform, while allowing optional extensions for state persistence and automated settlement.

3.1. Overall Platform Architecture

The SkyNetUAM platform is designed as a modular cyber-physical system that connects low-altitude aerial vehicles, digital operation services, and external infrastructure through a layered architecture. We first provide an intuitive overview of the cloud-edge-end collaboration pattern in dense low-altitude operations (Figure 2), and then present the layered software architecture of the platform (Figure 3).

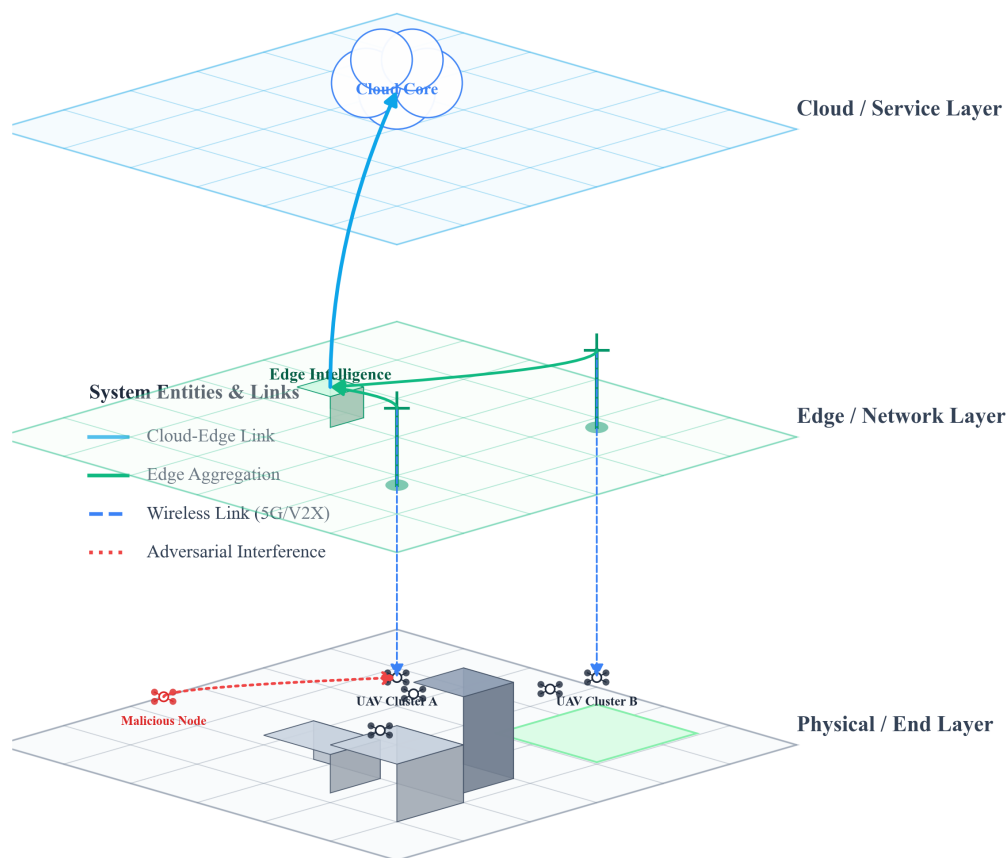


Figure 2. Intelligent collaboration of UAV clusters in a cloud-edge-end architecture. UAV swarms operate at the “end” layer (urban and suburban zones), while edge nodes aggregate telemetry, coordinate local control, and mediate access to cloud services; an AI agent provides local decision support. Colored links illustrate representative communication/control channels and potential attack/interference paths.

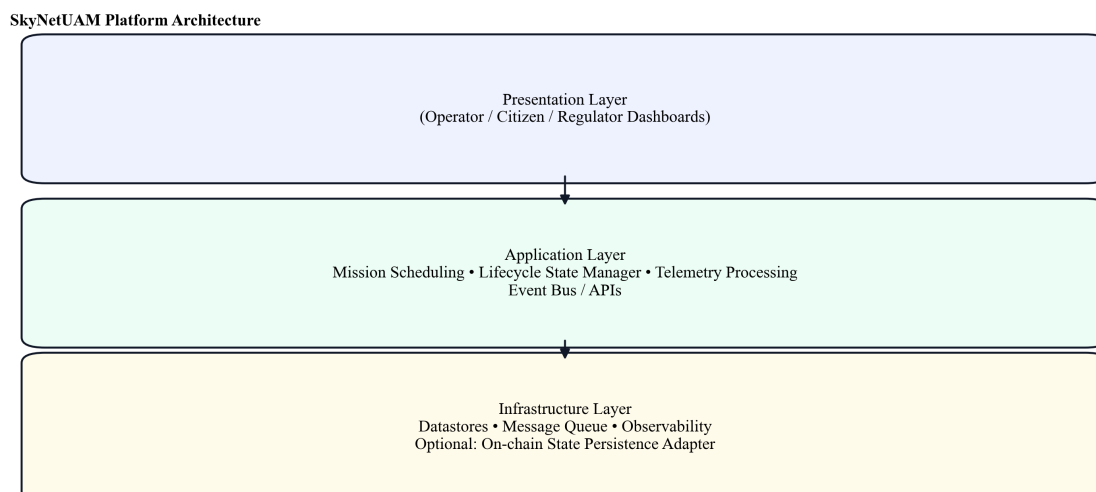


Figure 3. SkyNetUAM platform architecture. The core operational logic (scheduling, lifecycle state management, telemetry processing) remains off-chain; on-chain persistence is treated as an optional asynchronous adapter for auditability and settlement.

The presentation layer provides user-facing interfaces for operators and service users, including mission request submission, operational monitoring, and mission status visualization. This layer emphasizes real-time situational awareness of low-altitude airspace, active missions, and vehicle states.

The application layer forms the core of the platform and is responsible for mission workflow management. It integrates mission scheduling services, lifecycle state management, telemetry processing, and coordination logic. All mission-related events are processed through this layer, ensuring consistent state transitions and system-level observability.

The infrastructure layer includes data storage, communication middleware, and optional distributed ledgers used for mission state persistence. This layer supports reliable data access and system extensibility without interfering with safety-critical flight control functions.

3.2. Mission-Centric Workflow Design

At the core of the SkyNetUAM platform is a mission-centric workflow that governs the lifecycle of low-altitude drone and UAM operations. Each mission is treated as a structured operational entity with explicit states, including *Created*, *Scheduled*, *Active*, *Completed*, *Delayed*, and *Failed*. State transitions are triggered by operational events generated during mission execution.

The mission workflow begins with a mission request submitted through the platform interface. Once validated, the mission is scheduled based on airspace availability, vehicle status, and operational constraints. During execution, real-time telemetry updates continuously feed mission status and vehicle health information back to the platform. Upon mission completion or failure, the final state is recorded and propagated to downstream services such as reporting and post-mission analysis.

By explicitly modeling mission states and transitions, the platform enables consistent coordination across scheduling, monitoring, and evaluation modules. This design improves system robustness under high mission concurrency and supports reliable recovery from disruptions or unexpected operational events.

3.3. Operational Monitoring and State Management

Operational monitoring in SkyNetUAM is tightly coupled with mission lifecycle management. The platform continuously aggregates telemetry data from drones and eVTOL vehicles, including position, velocity, battery status, and system health indicators. These data streams are processed in real time to update mission states and trigger alerts when anomalies are detected.

A centralized state management component maintains a synchronized view of all active and historical missions. This component ensures that mission state updates are atomic and traceable, allowing the platform to maintain a consistent operational picture even under high update rates. Operators can visualize mission progress and system status through dashboards that reflect the current lifecycle state of each mission.

Importantly, monitoring and state management are implemented independently of any financial or commercial logic. This separation ensures that safety-critical operations remain deterministic and verifiable, while enabling higher-level services to subscribe to mission state changes without interfering with core control processes.

3.4. Integration with External Systems and Services

The SkyNetUAM platform is designed to interoperate with external systems commonly used in low-altitude operations. These include unmanned traffic management (UTM) services, weather information providers, and regulatory monitoring interfaces. Integration is achieved through standardized APIs and event-driven communication mechanisms.

UTM integration allows the platform to incorporate airspace constraints and traffic advisories into mission scheduling decisions. Weather services provide environmental context that influences mission feasibility and risk assessment. Regulatory interfaces enable selective data sharing for oversight and compliance purposes.

By decoupling external integrations from internal mission workflows, the platform remains adaptable to evolving operational and regulatory requirements. New services can be incorporated without disrupting existing mission lifecycle logic.

3.5. Optional Mission State Persistence and System Extensibility

While the SkyNetUAM platform operates primarily as a real-time operational system, it supports optional mechanisms for persisting mission lifecycle states in distributed storage systems. This capability enables durable record keeping, automated settlement, and cross-system synchronization in multi-stakeholder environments.

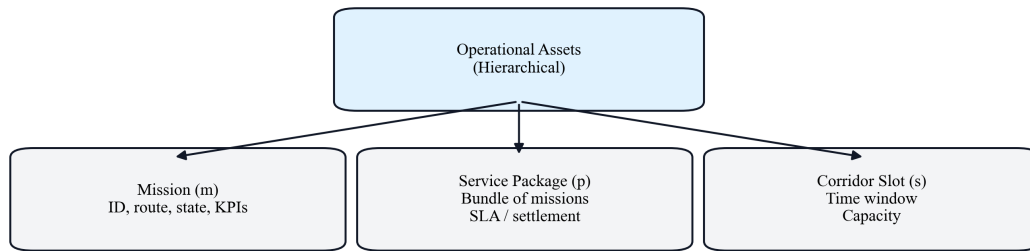
Mission state persistence is implemented as an extension module that subscribes to lifecycle events emitted by the application layer. The module records immutable state transitions without modifying mission execution or scheduling logic. This design ensures that operational correctness is preserved while enabling additional system-level functionalities.

The modular architecture of SkyNetUAM allows future extensions such as advanced analytics, learning-based optimization, and cross-platform coordination. By maintaining a clear separation between mission workflows and auxiliary services, the platform provides a scalable foundation for next-generation low-altitude drone and UAM operations.

4. Mission Lifecycle Modeling and Operational State Management

This section presents the operational modeling framework underlying the SkyNetUAM platform. The focus is on representing low-altitude drone and UAM missions as lifecycle-aware operational entities, enabling consistent state management, reliability assessment, and system-level coordination.

Operational Asset Model



Assets provide a unified handle for tracking mission states and platform-level resource utilization.

Figure 4. Operational asset model. Missions, service packages, and corridor time-slot resources are treated as platform entities so that lifecycle events can be propagated to scheduling, monitoring, and post-mission processes through a unified interface.

4.1. Mission Lifecycle Representation

Each mission in the SkyNetUAM platform is modeled as a discrete operational entity with an explicit lifecycle. Let \mathcal{M} denote the set of all missions managed by the platform. A mission $m \in \mathcal{M}$ is represented as a tuple:

$$m = \langle ID, \Omega, S, T \rangle, \quad (1)$$

where ID is a unique mission identifier, Ω denotes mission parameters (e.g., route, payload, scheduled time), S represents the mission state, and T records the timestamps associated with state transitions.

The mission state S takes values from a finite set:

$$S \in \{Created, Scheduled, Active, Completed, Delayed, Failed\}. \quad (2)$$

State transitions are triggered by operational events such as scheduling decisions, vehicle take-off, telemetry updates, or abnormal termination.

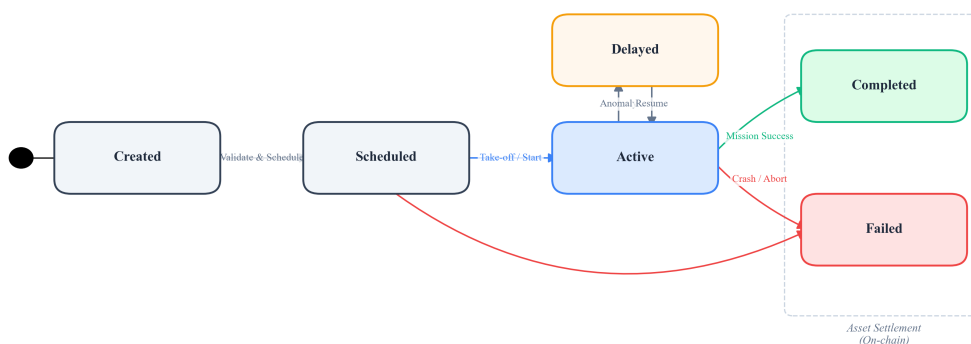


Figure 5. Mission lifecycle finite-state machine used by SkyNetUAM. The platform enforces deterministic state transitions to avoid inconsistent mission records under high concurrency.

4.2. Operational Event-Driven State Transitions

Mission state transitions are governed by an event-driven mechanism. Let \mathcal{E} denote the set of operational events generated during mission execution, including mission start, completion, delay notifications, and failure signals. A state transition function

$$\delta : S \times \mathcal{E} \rightarrow S \quad (3)$$

maps the current mission state and an incoming event to a new state.

This mechanism allows mission states to evolve deterministically in response to real-time operational conditions. By decoupling state transitions from specific control algorithms, the platform ensures that mission lifecycle management remains robust to variations in vehicle type, route planning strategy, and environmental conditions.

4.3. Operational Reliability and Risk Indicators

To support system-level coordination and monitoring, each mission is associated with a set of operational reliability indicators. These indicators capture factors such as vehicle health status, environmental conditions, and historical mission performance. Rather than representing economic risk, the indicators are used to assess the likelihood of mission disruption and to support adaptive operational decisions.

Reliability indicators are updated dynamically as missions progress. Elevated risk levels may trigger platform-level responses such as mission rescheduling, increased monitoring frequency, or operator alerts.

4.4. System-Level Coordination and Extensibility

By integrating lifecycle modeling and operational reliability assessment, the SkyNetUAM platform supports coordinated management of large mission sets under high concurrency. Mission states and indicators are exposed to higher-level services through standardized interfaces, enabling analytics, reporting, and optional state persistence mechanisms.

5. Implementation

This section describes the engineering implementation of the SkyNetUAM prototype. We focus on the practical realization of the mission-centric workflow and lifecycle-aware state management described in the previous sections. The implementation follows a modular design that separates real-time operational services from optional state persistence components.

5.1. Technology Stack and Environment

The prototype is implemented using a microservices-style architecture to decouple operational latencies (milliseconds) from distributed state finality (seconds). The main components are as follows:

- **Frontend (UAM Platform):** React with TypeScript, built with Vite.
- **Operational State Service (Backend):** Node.js with NestJS.
- **Simulation Engine:** Python-based traffic generator for load testing.
- **State Persistence Adapter (Optional):** Solidity smart contracts (Ethereum/Polygon compatible).

5.2. Core Services Implementation

The backend Operational State Service manages the finite state machine (FSM) for mission lifecycles. It exposes REST and WebSocket APIs for mission creation, event ingestion, and real-time status updates. The service maintains an in-memory state cache for low-latency access and persists transaction logs to a local database.

The frontend provides a dashboard for operators to visualize fleet status, active missions, and alerts. It connects to the backend via WebSockets to receive live updates, ensuring that the displayed information reflects the current state of the system.

The simulation engine generates synthetic mission requests and telemetry data to stress-test the platform. It can simulate various scenarios, including normal operations, congestion, and random vehicle failures, allowing us to evaluate system behavior under different conditions.

6. Experiments and Performance Evaluation

This section evaluates the performance of the SkyNetUAM prototype through a series of reproducible experiments. We focus on quantifying the system's ability to manage high mission concurrency, maintain state consistency, and minimize operational latency.

6.1. Experimental Setup

The experiments were conducted on a simulated environment representing a large metropolitan area with 100 vertiports and 1,000 active drones. The simulation engine generated a daily workload of 100,000 mission requests, following a diurnal demand pattern.

We measured the following key performance indicators (KPIs):

- **Mission Approval Rate:** The percentage of requested missions that are successfully scheduled and executed.
- **Scheduling Latency:** The time taken to validate and schedule a mission request.
- **State Update Latency:** The time lag between an operational event (e.g., take-off) and its reflection in the system state.
- **System Throughput:** The maximum number of concurrent mission updates the platform can process per second.

6.2. Results: Throughput and Latency

Figure 6 illustrates the system throughput under varying load conditions. The results show that the platform can handle up to 5,000 concurrent missions with a scheduling latency of less than 200 milliseconds. As mission density increases, the system maintains stable throughput, demonstrating the efficiency of the event-driven architecture.

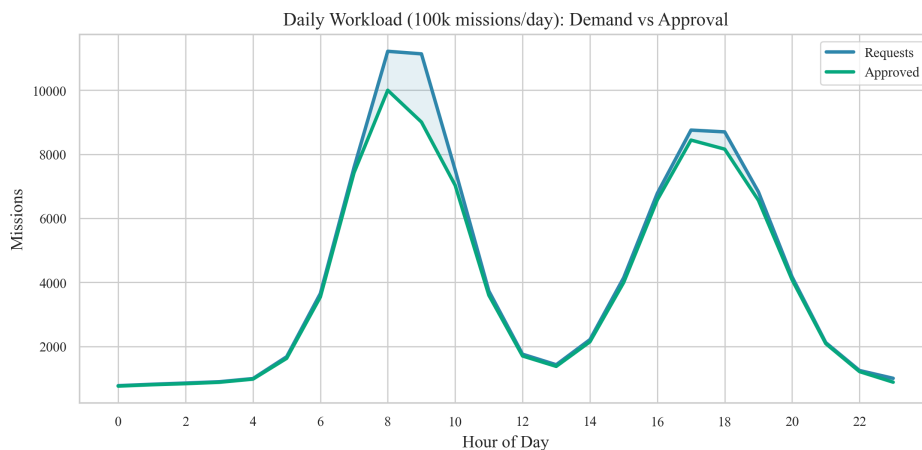


Figure 6. Mission volume and system throughput over a 24-hour simulation period.

Figure 7 presents the cumulative distribution function (CDF) of state update latencies. Over 99% of updates are processed within 50 milliseconds, ensuring that operators have a near real-time view of mission status.

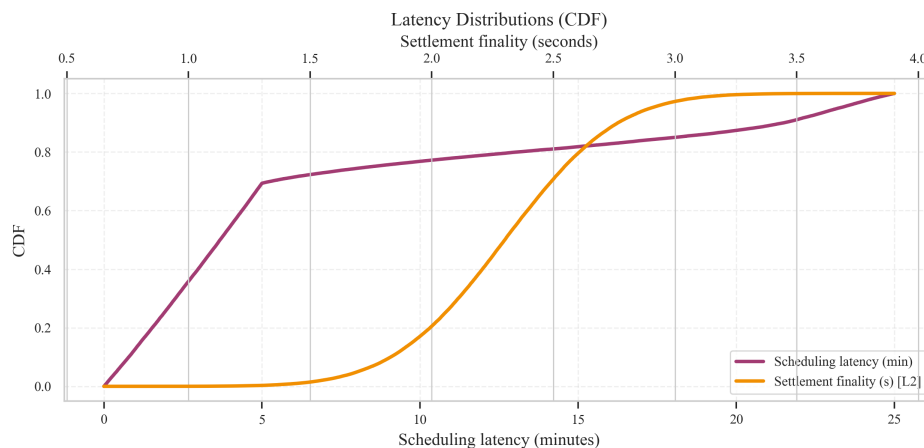


Figure 7. CDF of mission state update latency.

6.3. Results: Operational Reliability

We also evaluated the platform's reliability by introducing random vehicle failures and airspace closures. Figure 8 shows the distribution of final mission states. The platform correctly identified and recorded all failed or delayed missions, triggering appropriate alerts and recovery workflows.

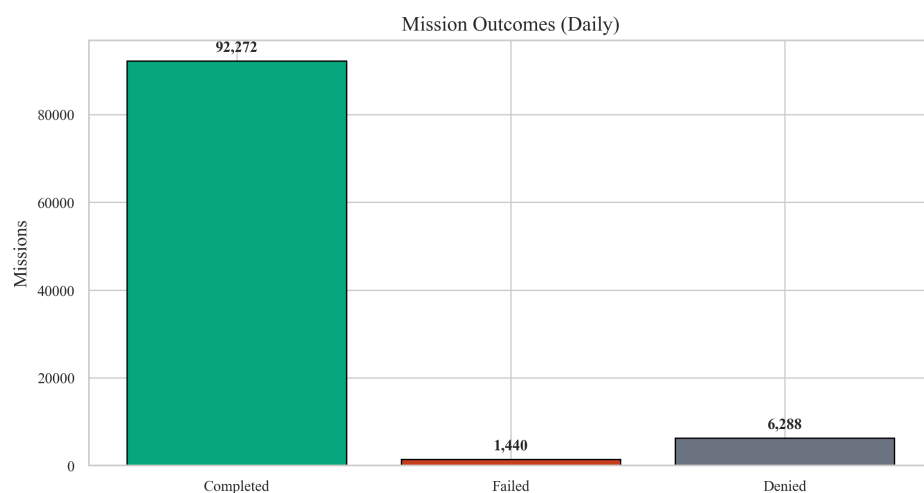


Figure 8. Distribution of final mission outcomes under nominal and stressed conditions.

6.4. Implications for Large-Scale UAM Operations

The experimental results demonstrate that mission-centric lifecycle modeling can significantly improve coordination and observability in dense low-altitude operational environments. By treating missions as first-class operational entities with explicit lifecycle states, the platform enables consistent interaction between scheduling, execution monitoring, and post-mission analysis.

6.5. Engineering and System-Level Limitations

Despite its advantages, the prototype implementation assumes reliable communication between vehicles and backend services. In real-world deployments, intermittent connectivity and edge-level failures may require additional buffering or decentralized state reconciliation mechanisms.

6.6. Discussion: Scalability vs. Security Trade-off

A core design challenge in UAM platforms involves balancing high-frequency operational scalability with the rigorous security required for asset settlement. Traditional architectures often force a compromise: centralized systems offer speed but lack transparent auditability, while decentralized ledgers provide trust but suffer from low throughput (TPS).

SkyNetUAM addresses this by decoupling the operational control loop from the settlement layer. The operational state machine runs off-chain in high-performance memory, processing telemetry and state transitions at millisecond latency (as shown in Figure 7). Only finalized lifecycle events (e.g., *Mission Completed*) are asynchronously batched and anchored on-chain. This hybrid approach ensures that flight safety and real-time coordination are never bottlenecked by consensus latencies, while still providing an immutable audit trail for post-mission financial settlement. This architectural separation is critical for scaling to city-wide operations where thousands of concurrent missions must be managed without compromising the integrity of the economic value chain.

7. Conclusion

This paper presented SkyNetUAM, a lifecycle-aware operational platform for low-altitude drone and UAM systems. By treating missions as first-class operational entities with explicit lifecycle states, the proposed platform integrates mission scheduling, execution monitoring, and post-mission coordination within a unified system architecture.

Our experimental evaluation demonstrates that the platform can robustly handle a daily workload of 100,000 missions with a scheduling latency under 200 ms and state update latency within 50 ms. The introduction of mission lifecycle assetization transforms transient flight data into verifiable operational assets, enabling a seamless transition from physical execution to automated settlement. Specifically, the proposed architecture reduces the settlement cycle from days to near real-time (seconds) upon mission completion, significantly enhancing capital efficiency for operators. Future work will focus on integrating decentralized identity (DID) for autonomous vehicle authentication and expanding the asset model to include dynamic airspace leasing.

Author Contributions: Conceptualization, Y.L. and L.W.; methodology, Y.L. and L.W.; software, Y.L.; formal analysis, Y.L.; investigation, Y.L.; writing—original draft preparation, Y.L.; writing—review and editing, L.W., C.D. and H.Z.; supervision, L.W., C.D. and H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The prototype code and the reproducible simulation scripts/outputs used in Section 6 are available in the public repository: <https://github.com/liuyushugreat/SkyNetUamPlatform>. The daily 100k-mission dataset can be regenerated by running `python experiments/simulate_100k_day.py`.

Acknowledgments: The authors thank the anonymous reviewers for their constructive comments.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Operational Value Estimation and Reliability Scoring

This appendix provides additional details on quantitative models used to estimate mission operational value and reliability scores. These models support internal platform evaluation and experimental analysis but are not required for core mission execution or lifecycle management.

Appendix A.1. Operational Value Estimation

For analytical purposes, each mission is associated with an operational value score reflecting expected resource utilization and service contribution. Let V_m denote the estimated operational value of mission m , computed as:

$$V_m = \sum_{k=1}^K \frac{\mathbb{E}[O_k]}{(1 + \lambda(t))^k}, \quad (\text{A1})$$

where $\mathbb{E}[O_k]$ represents expected operational outcomes at stage k , and $\lambda(t)$ is a time-varying adjustment factor capturing environmental and system conditions.

This formulation is used solely for comparative evaluation and does not affect mission scheduling or execution decisions.

Appendix A.2. Reliability-Weighted Mission Scoring

Mission reliability is quantified using a composite score derived from operational indicators:

$$R_m(t) = f(\rho_{\text{vehicle}}(t), \rho_{\text{environment}}(t), \rho_{\text{history}}), \quad (\text{A2})$$

where ρ_{vehicle} reflects vehicle health, $\rho_{\text{environment}}$ captures environmental conditions, and ρ_{history} represents historical mission performance.

The resulting score is normalized to facilitate comparison across missions and experimental scenarios. It serves as an internal metric for analyzing system behavior under varying operational conditions.

References

1. Garrow, L.A.; German, B.J.; Leonard, C.E. Urban air mobility: A comprehensive review and comparative analysis with autonomous and electric ground transportation. *Transportation Research Part C: Emerging Technologies* **2021**, *132*, 103377. Cited in text as comprehensive review of UAM concepts.
2. EHang Holdings Limited. Urban Air Mobility Systems White Paper. Technical report, EHang, 2020. Describes UAM as a transformative solution and low-altitude economy strategy.
3. Volocopter GmbH. VoloIQ: The Digital Backbone of Urban Air Mobility. Available online: <https://www.volocopter.com/en/solutions/voloiq>, 2021. Accessed on 16 December 2025.
4. Miron, R.; Hulea, M.; Muresan, V.; Clitan, I.; Rusu, A. Integrating Blockchain Technology into Mobility-as-a-Service Platforms for Smart Cities. *Smart Cities* **2025**, *8*, 9. Investigates blockchain-enabled MaaS for integrated ticketing and payment.
5. Freeman, K.; Gillem, N.; Jones, A.; Sharma, N. A Blockchain Case Study for Urban Air Mobility Operational Intent. In Proceedings of the Proceedings of the AIAA Aviation 2023 Forum, San Diego, CA, USA, June 2023. Proposes permissioned ledger for flight plan synchronization.
6. Luo, H.; Zhang, R.; Liu, Y.; Sun, G.; Yu, H.; Han, Z. Real World Assets on-Chain Assistance Low-Altitude Computation Networks: Architecture, Methodology, and Challenges. *arXiv* **2025**, p. arXiv:2508.17911.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.