RESEARCH ARTICLE

Edge-Cloud Architectures for Urban Mobility and Safety

[version 1; peer review: 2 approved with reservations]

Enrico Rossini[1], Marcello Pietri[1], Marco Picone[1], Natalia Selini Hadjidimitriou[1], Marco Mamei[1], Panagiotis Moraitis[2], Petros Zervoudakis[2]

[1]DISMI, University of Modena and Reggio Emilia, Reggio Emilia, Italy
[2]Netcompany, Athens, Greece

v1 **First published:** 06 Jan 2026, **6**:14
https://doi.org/10.12688/openreseurope.22191.1
Latest published: 06 Jan 2026, **6**:14
https://doi.org/10.12688/openreseurope.22191.1

Abstract

This paper introduces the NOUS Smart City Architecture (NSCA), an extensible middleware designed to enable intelligent, interoperable urban services across the edge–cloud continuum. Developed within the NOUS project, NSCA addresses a central challenge in urban digitalization: supporting scalable, efficient communication among heterogeneous actors while integrating with emerging data space infrastructures. On the **edge side**, NSCA builds on the lightweight and widely adopted MQTT protocol, enabling reliable, low-latency, and semantically structured communication among distributed assets such as edge devices, vehicles, roadside units, and localized digital services. On the **cloud side**, it integrates a Data Space Ecosystem based on the SIMPL Open architecture, providing a scalable foundation for cross-stakeholder data exchange and governance. The two layers are bridged through a standard MQTT inter-broker connector that can be deployed on either edge nodes or cloud hosts, offering flexibility in how topics are forwarded, filtered, and federated. NSCA supports both real-time data dissemination and service composition at scale, allowing new applications to be integrated with minimal configuration. This makes the architecture suitable for a wide range of smart city verticals, including road safety, mobility management, environmental monitoring, and citizen services. The platform's practical value is demonstrated through two operational services deployed in a real-world smart city testbed. The **Vulnerable Road User (VRU) Safety Service** leverages edge-to-cloud communication to detect pedestrians, cyclists, and other users at risk, issuing proactive collision warnings based on real-time situational awareness. The **GeoPerception Service** enables cooperative sensing by sharing localized perception among vehicles and smart infrastructures, improving environmental understanding and mobility intelligence. Together, these services showcase NSCA's ability to

Open Peer Review

Approval Status ? ?

	1	2
version 1 06 Jan 2026	? view	? view

1. **Shiva Shankar Reddy**, Sagi Rama Krishnam Raju Engineering College (A), Bhimavaram, India

2. **Eirini Eleni Tsiropoulou**, Arizona State University, Tempe, USA

Any reports and responses or comments on the article can be found at the end of the article.

integrate IoT data flows, edge analytics, and data-space–enabled cloud services, supporting next-generation, interoperable urban applications.

Keywords
Edge computing, Cloud computing, Multi-Access Edge Computing, Mobility, Safety, Smart City, Data Spaces

This article is included in the Horizon Europe gateway.

Corresponding author: Marco Mamei (marco.mamei@unimore.it)

Author roles: Rossini E: Conceptualization, Methodology, Software, Writing – Review & Editing; **Pietri M**: Conceptualization, Software, Writing – Review & Editing; **Picone M**: Conceptualization, Software, Supervision, Writing – Review & Editing; **Hadjidimitriou NS**: Conceptualization, Funding Acquisition, Investigation, Project Administration, Writing – Review & Editing; **Mamei M**: Conceptualization, Funding Acquisition, Investigation, Project Administration, Supervision, Writing – Original Draft Preparation, Writing – Review & Editing; **Moraitis P**: Conceptualization, Funding Acquisition, Project Administration, Writing – Review & Editing; **Zervoudakis P**: Conceptualization, Project Administration, Software, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101135927 (A catalyst for EuropeaN ClOUd Services in the era of data spaces, high-performance and edge computing - NOUS)
The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

How to cite this article: Rossini E, Pietri M, Picone M *et al.* **Edge-Cloud Architectures for Urban Mobility and Safety [version 1; peer review: 2 approved with reservations]** Open Research Europe 2026, **6**:14 https://doi.org/10.12688/openreseurope.22191.1

First published: 06 Jan 2026, **6**:14 https://doi.org/10.12688/openreseurope.22191.1

Introduction

The rapid evolution of modern cities into complex cyber-physical ecosystems has created the need for two different classes of services. On the one hand, there are real-time low-latency services associated with road safety, autonomous driving, real time communications, etc. On the other hand, there are complex multi-stakeholder applications focusing on heterogeneous data exchange and analysis: public services, energy optimization, e-governments, etc. Therefore, the smart city infrastructure should support such often contrasting requirements. Physical assets, mobile users, and digital services are increasingly required to interact seamlessly across distributed environments, generating and consuming large volumes of data under strict latency and reliability constraints. At the same time, stakeholders must engage in complex interaction protocols that involve heterogeneous and large-scale data sources.

Within this scenario, the NOUS framework aims at creating a novel architecture for edge-cloud services supporting low-latency applications as in smart cities and safety scenarios that extends to trusted, regulated and complex-multi party interactions via cloud data exchange.

We applied the NOUS framework to a concrete real-world smart city testbed: the Modena Automotive Smart Area (MASA). MASA is a living laboratory for smart urban innovation. Located in Modena, Italy, MASA integrates diverse connected assets—including Smart Cameras, Roadside Units (RSUs), vehicular on-board units, and personal mobile devices. These components operate across multiple layers of an edge–cloud continuum, ranging from low-power edge devices to Multi-Access Edge Computing (MEC) nodes and federated Cloud services, enabling large-scale experimentation with cooperative, data-driven services for smart cities.

However, managing communication and coordination in such a heterogeneous, dynamic, and distributed environment poses significant challenges. Systems must remain modular and evolvable while preserving interoperability, scalability, and low-latency performance. To address these challenges, this paper introduces the NSCA, a flexible, event-driven middleware designed to support intelligent interactions among heterogeneous urban entities distributed along the edge–cloud continuum. In particular, it supports low-latency event-based communication on the edge side, and it supports data exchange through data spaces among multiple stakeholders on the cloud side.

More in detail, NSCA revolves around two key principles:

On the edge side, one of the key architectural principles of NSCA is logical decoupling between data producers and consumers, enabling systems and services to evolve independently. To achieve this, NSCA adopts an event-driven paradigm based on the Publish/Subscribe (Pub/Sub) communication model, which naturally supports asynchronous, many-to-many interactions among distributed components. This model promotes high modularity, fault tolerance, and responsiveness, making it particularly suitable for latency-sensitive scenarios, where computation must often be executed close to data sources at the edge, while still enabling integration with cloud analytics and decision-making components. NSCA implements this paradigm through a structured messaging layer that organizes interactions into semantic categories such as telemetry, events, actions, and service-level exchanges. This semantic structuring provides fine-grained control over data flows, allowing selective subscriptions, hierarchical topic organization, and dynamic service discovery mechanisms. Furthermore, NSCA adopts an edge–cloud broker deployment strategy, enabling local message routing and processing for time-critical services, while preserving interoperability with remote observers, data aggregators, and third-party services.

On the cloud side, NCSA adopts the Data Space framework as a core architectural principle to enable secure, scalable, and sovereign data sharing across organizations. A Data Space is a distributed and federated ecosystem that allows independent stakeholders to exchange data without relinquishing control over their assets, while ensuring transparency and contractual clarity. In contrast to centralized approaches based on data lakes or data warehouses, a Data Space preserves data sovereignty by keeping data close to its original owner and enabling controlled access through standardized policies, rather than physical data consolidation. To support this paradigm, NCSA integrates interoperability mechanisms at multiple levels—syntactic interoperability via shared APIs and communication protocols; semantic interoperability through standardized data models and ontologies; and organizational interoperability enforced by governance rules and data usage agreements. Trust is established through distributed identity management, certification mechanisms, and usage control frameworks that regulate how data can be accessed, reused, and combined into value-added services. Furthermore, NCSA enables federated data processing and AI workflows, allowing cloud services to operate on distributed datasets while respecting access permissions and usage constraints. This approach facilitates cross-domain integration—such as mobility, safety, logistics, energy and sustainability—while aligning with European initiatives like GAIA-X and the Simpl-Open, which promote openness, transparency, and digital sovereignty in data ecosystems.

The combination of these two layers allow to create complex multi-stakeholder applications for smart city data, while retaining flexibility and low-latency at the edge level to support real-time applications.

This paper provides a detailed description of the NSCA communication model, along with its topic structure, interaction semantics, and deployment strategy. The architecture is validated through two real-world services deployed within MASA:

- The Vulnerable Road User (VRU) Safety Service, which detects and prevents collision risks involving pedestrians and cyclists by leveraging cooperative sensing and real-time alerts
- The GeoPerception Service, which combines data from heterogeneous sensors to deliver localized environmental awareness and statistics on city mobility to infrastructure and users.

These services demonstrate how NSCA supports scalable, event-driven coordination and enables the deployment of intelligent, data-driven smart city applications that can enhance safety, mobility efficiency, and situational awareness in complex urban environments.

This paper is organized as follows: Section 2 presents the main NSCA edge components. Section 3 presents the main NCSA cloud components. Section 4 presents reference applications. Section 5 presents related work in the area. Section 6 concludes and presents future work.

Nous Smart City Architecture (NSCA) - Edge

The NSCA-edge is engineered to support real-time, flexible, and scalable communication among distributed components operating within complex urban environments. Its design combines a layered, event-driven messaging system with a hybrid deployment strategy that integrates Edge and Cloud brokers. This hybrid approach enables low-latency local computation for time-critical services, while simultaneously supporting wide-area data dissemination for monitoring, analytics, coordination, and remote service provisioning.

At the core of NSCA lies a distributed event-driven communication fabric that enables asynchronous, decoupled interactions among heterogeneous clients. These include:

- Physical assets, such as smart cameras, roadside units, and mobile devices;
- Software services, including perception modules, risk detection engines, and decision-support logic;
- External observers, such as dashboards, data aggregators, and supervisory control systems.

Although the architecture is designed to be protocol-agnostic and technology-independent, the current prototype instantiates NSCA using the MQTT (Message Queuing Telemetry Transport) protocol due to its lightweight footprint, reliability over unreliable networks, and wide adoption in IoT ecosystems. Communication among these entities follows the publish/subscribe (Pub/Sub) model, which promotes logical decoupling between message producers and consumers. This paradigm supports system modularity, elastic scalability, service composability, and resilience to dynamic network conditions, all of which are necessary for real-time smart city applications.

Hybrid edge–cloud deployment

To meet heterogeneous Quality of Service (QoS) requirements, SCA employs a multi-broker architecture:

- *Edge Broker* – Deployed within local domains (e.g., MEC nodes or micro data centers), it handles latency-critical communication, allowing localized interactions among nearby assets and services. This is essential for safety-critical use cases, such as real-time Vulnerable Road User (VRU) risk alerts or cooperative perception.
- *Cloud Broker* – Deployed in the public cloud or institutional data centers, it enables global service access, long-term storage, and support for cross-domain applications, including fleet monitoring, digital twin synchronization, and historical analytics. It connects the Edge part with NSCA-cloud and data spaces (see next Section).

To ensure transparent interoperability between these two layers, brokers are connected through a bridge mechanism. This allows selective message replication, topic forwarding, and cross-domain routing, enabling data to remain local by default while being escalated to the cloud when necessary for supervision or large-scale aggregation. In practical terms, the bridge operates as a native MQTT inter-broker link that forwards only the topics explicitly configured for propagation. Each broker maintains its own namespace, and the bridge maps selected topic trees from the edge broker to the cloud broker using configurable include/exclude rules. This setup prevents full-mesh synchronization and ensures that only the required subtopics—such as aggregated events, service outputs, or supervisory metadata—are transmitted upstream. The mechanism also supports bidirectional forwarding, allowing cloud-based services to publish commands or configuration updates that are injected into the edge broker with controlled scope. By relying on MQTT's built-in session handling, QoS policies, and retained messages, the bridge guarantees consistent delivery semantics across domains without modifying client applications. This enables scalable, selective routing between brokers while keeping the messaging load tightly bounded at the edge.

A distinguishing feature of NSCA is its semantic topic hierarchy, which structures message flows according to functional roles and data scope. This design enables: (i) Fine-grained topic filtering for efficient message routing; (ii) Service discovery via structured topic introspection; (iii) Action invocation and response routing based on command semantics; (iv) Logical segregation of communication categories (telemetry, events, commands, and service responses). To ensure consistency and traceability in dense deployments, NSCA enforces naming conventions and message metadata standards. Each asset is uniquely referenced using `<asset_type>/<asset_id>`, while services register using standardized descriptors. Action-response interactions are tracked using request identifiers (request_id) to maintain reliability in asynchronous command execution. Together, these design choices enable NSCA to serve as a modular, evolvable, and MQTT-native middleware tailored to the communication demands of smart urban ecosystems, supporting thousands of concurrent entities distributed across heterogeneous networks.

Client categorization and interaction model

To organize communication and interaction within the NSCA ecosystem, all connected entities are categorized into three functional client roles, based on their operational purpose and interaction semantics:

1. *Assets* – These are NSCA clients representing physical or logical entities that produce data streams or accept control commands. Typical examples include environmental sensors, wearable devices, connected vehicles, mobile applications, and smart cameras. Each asset is uniquely identified using an `<AssetType>/<AssetId>` convention, which ensures consistent naming and traceability across deployments. Assets primarily publish

telemetry, status information, and event notifications, and may subscribe to action commands issued by services.

2 *Services* – Services are active computational modules that process incoming data, implement operational logic, and generate outputs for consumption by other system participants. SCA distinguishes between two types of services:

   - Generic Services, which are openly accessible without prior registration (e.g., weather information, map services);
   - Asset Services, which require explicit registration per assets and provide context-aware, client-specific outputs, such as personalized VRU safety alerts or cooperative perception feedback. Services publish their availability and interaction contracts through dedicated topic channels, enabling dynamic discovery and on-demand interaction.

3. *Observers* – Observers are read-only clients that subscribe to specific topic hierarchies for monitoring, auditing, or visualization purposes. Examples include control room dashboards, system supervisors, digital twin interfaces, and logging tools. Though they do not actively influence system behavior, observers play a crucial role in situational awareness and system transparency.

NSCA supports multi-role participation, allowing a single application to assume multiple roles concurrently. In practice, an application may act as an asset when publishing its own telemetry, operate as a service when processing data for other clients, and simultaneously function as an observer by subscribing to selected topics for monitoring or analytics. This flexibility also enables asymmetric interactions: an application can be an asset from the perspective of one service while serving as a computation module or observer for others, facilitating more dynamic and modular system compositions.

Communication among these entities is orchestrated through the NSCA broker infrastructure, which provides message routing, service coordination, and lifecycle management. Supported interaction patterns include:

- *Service Registration* – Assets register to Asset Services through dedicated registration topics, enabling per-client service sessions.
- *Service Publication* – Service availability and metadata are published as retained information messages, allowing late joiners to discover active services.
- *Service Notification* – Services publish tailored outputs to asset-specific channels (e.g., risk_level for VRU alerts).
- *Action Invocation* – Assets and observers may invoke commands on services, with optional request/response correlation via unique identifiers.

These message flows are illustrated in Figure 1, which depicts the information exchange across the MASA Edge–Cloud Continuum, showing how assets, services, and observers interact via telemetry streams, events, actions, service discovery, and notifications across edge and cloud brokers. The left part of the figure shows the edge–cloud bridge mechanism connecting the edge broker and the cloud broker, enabling selected topics to be forwarded across domains while keeping most traffic local to the edge. The right part illustrates the interaction between assets and services—covering service discovery, registration, and event exchange—as well as the observer role, which can subscribe to asset telemetry and service notifications to monitor the system without affecting its operation.

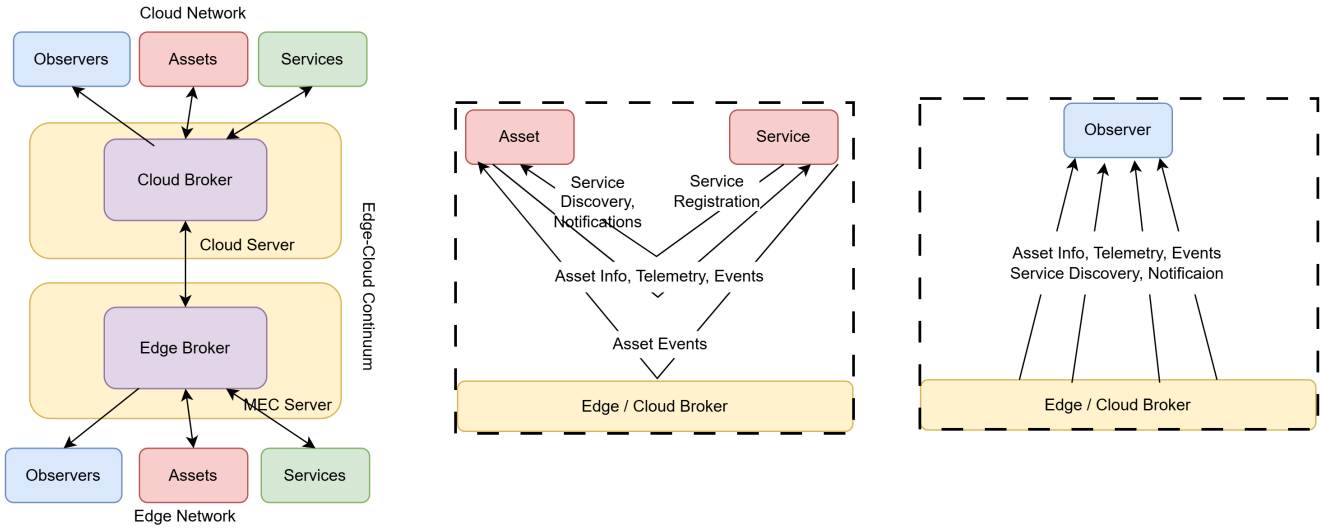

Figure 1. Edge Cloud broker in NOUS. The left side shows the edge–cloud bridge linking edge and cloud brokers for selective topic forwarding. The right side illustrates asset–service interactions and the observer role, which monitors telemetry and notifications without influencing operations.

Message topic design and flow model

The NSCA communication fabric is organized around a semantic topic hierarchy that structures message flows according to functional categories such as telemetry, events, actions, and service coordination. Topics follow a hierarchical naming convention that separates concerns while enabling efficient routing and interoperability. Although natively implemented using MQTT, the topic conventions are technology-agnostic and can be mapped to other event-driven messaging solutions—such as ZeroMQ or AMQP—through routing key translation. Additional subtopics enable resource-level filtering, allowing clients to subscribe only to relevant data without processing unnecessary streams. This organization results in bandwidth-efficient communication and supports fine-grained access control, both essential for large-scale smart city deployments.

The NSCA topic model also supports stateful and asynchronous interactions through: (i) Retained messages, which allow new clients to retrieve the latest state immediately upon subscription; (ii) Request/response subtopics, which enable service acknowledgements and correlation of asynchronous workflows. Payloads are encoded using JSON enriched with schema definitions and version tags, ensuring extensibility and backward compatibility. Each message carries structural metadata enabling consistent parsing, system observability, and message tracing across distributed brokers.

Together, the topic-driven architecture and broker-layer deployment strategy allow SCA to support high-frequency, low-latency communication required by mission-critical smart city services, such as real-time safety systems or cooperative perception, while preserving interoperability with external platforms and future integration with data space architectures and event-driven orchestration frameworks.

Dynamic plugin and filtering

To enhance flexibility and control over message handling within SCA, we introduced broker-level extensibility through custom plugins[1]. In the current implementation, an MQTT broker plugin written in C for performance-critical environments is employed. The plugin leverages libyaml to load a declarative YAML configuration, enabling system integrators to specify message transformation, filtering, and routing rules directly at the broker layer. These rules operate over both topic structure semantics and payload content, aligning with NSCA's architectural model. A key advantage of this solution is its runtime configurability: plugin behavior can be modified without recompilation or downtime, allowing deployment environments to adapt dynamically to new service requirements. By executing inside the broker's main event loop, the plugin introduces sub-millisecond processing latency, making it suitable for real-time and safety-critical smart city scenarios. Typical transformations supported by the plugin include:

- *Payload validation and sanitization* (schema compliance, field checking),
- *Metadata augmentation* (timestamp insertion, coordinate tagging, source labeling),
- *Conditional routing* based on content attributes (e.g., on the basis of asset type or geographic tags),
- *Protocol translation and republishing* to internal service topics.

For instance, in mobility use cases, a mobile application may publish GPS telemetry to a generic topic. The plugin can validate message fields, enrich the telemetry with standardized timestamps, classify the spatial region, and reroute the data to a service-specific topic (e.g., VRU risk evaluation) or trigger alert generation if malformed or anomalous patterns are detected.

A central requirement for scalability in large scale smart city systems is efficient filtering, especially in dense urban sensing environments where thousands of entities generate continuous data streams. Broker-side filtering ensures that only relevant events are forwarded to subscribed services, significantly reducing network load and processing overhead. This is particularly effective for spatially constrained services, which only require events within defined geographic regions or risk zones. The combination of hierarchical topic organization and plugin-driven filtering logic enables selective dissemination close to the data source, reducing latency and preventing unnecessary message propagation.

This modular and extensible approach empowers NSCA to incorporate cooperative edge intelligence without modifying client applications or services. It also creates a foundation for future extensions, such as edge-level anomaly detection, adaptive prioritization, local caching, and context-aware access control. By shifting part of the decision logic to the broker layer, NSCA enhances system responsiveness and scalability while maintaining interoperability and deployment agility.

Nous Smart City Architecture (NSCA) - Cloud

The transition toward next-generation smart city infrastructures increasingly depends on the ability of multiple organizations—public administrations, mobility operators, utility providers, private companies, and technology vendors—to exchange data in a trusted, controlled, secure, and transparent manner. Traditional cloud architectures, typically built around centralized data lakes or tightly coupled integration pipelines, no longer suffice in ecosystems where data sources are heterogeneous, distributed, and bound by strict governance constraints. To address these limitations, the European Union is promoting a new paradigm for cross-organizational collaboration: **Data Spaces**[2,3].

A **Data Space** is a federated, interoperable, and governance-driven ecosystem in which independent stakeholders can share data without relinquishing ownership or control. Instead of aggregating data in a single repository, each participant maintains sovereignty over its assets while exposing them through standardized interfaces, usage policies, and contractual frameworks. This approach aligns with the principles of transparency, trust, interoperability, and privacy-by-design that underpin the broader European strategy for digital sovereignty. In particular, Data Spaces are envisioned as foundational components for

EU-wide initiatives such as the European Mobility Data Space, the Smart Communities Data Space, and cross-domain environments that connect mobility, energy, logistics, tourism, and public administration services.

Within this strategic landscape, **SIMPL/Open (SIMPL)**—the *Smart Middleware for Trusted Data Spaces*—plays a crucial enabling role[4]. SIMPL is an open, modular, and standards-aligned middleware architecture developed by the European Commission to simplify the creation, deployment, and operation of trustworthy data spaces. It provides a reference set of building blocks that implement the technical foundations required for data governance, interoperability, identity management, and secure data exchange. Rather than prescribing a single monolithic platform, SIMPL promotes a highly composable architecture: each organization can adopt only the components it needs, while maintaining compatibility with other participants through common APIs, vocabularies, policy frameworks, and security services.

At its core, SIMPL supports three complementary goals:

1. *Data sovereignty and trust enforcement.* SIMPL incorporates mechanisms for declarative data usage policies, identity and credential management, auditing, and contract negotiation, ensuring that data providers can specify how their assets may be accessed, processed, or redistributed. These policies follow European values of privacy, security, and accountability, making SIMPL a natural fit for public-sector and multi-stakeholder scenarios typical of smart cities.

2. *Interoperability and semantic consistency.* The framework defines common data models, metadata vocabularies, and semantic reference structures that allow heterogeneous systems to communicate using shared meaning. This is essential in domains like mobility or urban management, where datasets originate from traffic sensors, IoT devices, administrative registries, digital twins, or third-party platforms.

3. *Federation rather than centralization.* SIMPL enables multiple autonomous domains to interconnect through a federated data infrastructure, in which each participant runs its own connectors, catalogues, and policy engines. This model supports scalable collaboration among cities, agencies, and private entities without requiring them to surrender data or depend on a single platform operator.

Integrating SIMPL/Open into the Nous Smart City Architecture (NSCA)

In the **NSCA Cloud layer**, the SIMPL/Open architecture serves as the foundation upon which the NOUS ecosystem builds its Data Space–compliant services. While the NSCA Edge layer handles real-time telemetry, perception, and low-latency coordination through MQTT-based publish/subscribe mechanisms, the NSCA Cloud layer focuses on multi-stakeholder data exchange, long-term storage, policy-driven interoperability, and cross-domain analytics. SIMPL/Open directly supports this by providing:

- *EDC-based data exchange pipelines*, enabling secure contract negotiation, data transfer, and controlled access to event-driven urban data streams.

- *Federated catalogs and metadata registries* that make data assets discoverable across multiple organizations while preserving semantic uniformity.

- *Common services for identity, authorization, and certification,* ensuring consistent trustworthiness and verifiable compliance across the federation.

- *Policy enforcement and auditing mechanisms* that regulate usage conditions, enabling providers to expose only approved subsets of their telemetry, events, or historical datasets.

- *A modular governance layer* that supports scalable involvement of city departments, utility operators, mobility companies, research partners, and private stakeholders.

In this model, the NOUS Data Space acts as the cloud-side extension of NSCA, enabling cities not only to share real-time data but also to create value-added services, cross-city analytics, and interoperable digital twins aligned with EU-wide Data Space initiatives. For example, mobility services running in MASA can be exposed—under clear contractual terms—to other municipalities, industrial partners, or national platforms; environmental data can be combined with traffic patterns; and multi-city intelligence can be developed without violating data protection constraints or duplicating storage.

For urban environments, Data Spaces and SIMPL/Open unlock several strategic capabilities:

- *Multi-stakeholder cooperation by design.* Cities can coordinate with transport operators, emergency responders, energy providers, and technology partners using unified, policy-governed data channels.

- *Cross-city federation.* Independent NSCA deployments across different municipalities can interoperate, aligning with EU digital policies.

- *Data sovereignty and GDPR compliance.* Sensitive data—including mobility traces, sensor telemetry, and video-derived metadata—remains under the control of its originator.

- *Scalable, interoperable digital twins.* Federated data exchange enables the creation of shared, cross-domain digital twins for mobility, environment, logistics, and public safety.

- *Reduced vendor lock-in.* SIMPL's open and modular architecture enables plug-and-play adoption of connectors, catalogs, and governance components compatible with the European Data Space ecosystem.

By integrating SIMPL/Open as the backbone of its Cloud layer, NSCA positions itself not only as an efficient smart city

middleware but also as a **future-proof, Data Space–aligned architecture**, capable of supporting advanced urban services across cities, regions, and national infrastructures. Building on these principles, the NSCA Cloud layer adopts SIMPL/Open as the foundation for implementing trusted, sovereign, and interoperable data exchange among smart-city stakeholders. While the introductory discussion highlights the strategic relevance of Data Spaces and the role of SIMPL in enabling secure federation, the following sections describe how these concepts are concretely instantiated within the NOUS architecture. In particular, NSCA integrates SIMPL/Open through a set of operational components that manage event-driven data, publish it as standardized data assets, negotiate access policies, and coordinate the transfer of information across organizational boundaries.

To make these mechanisms actionable in real deployments, NSCA structures its cloud-side workflow around two complementary perspectives: the **data provider**, responsible for exposing event-based data streams to the Data Space, and the **data consumer**, responsible for discovering, accessing, and integrating shared data assets. This separation reflects the federated nature of Data Spaces, where each participant maintains control over its resources while interoperating through common catalogs, connectors, and policy enforcement services (see Figure 2).

The next subsections describe these two roles in detail. They illustrate how raw events produced at the edge are synchronized, filtered, registered, and ultimately shared through the SIMPL/Open-compliant infrastructure; and conversely, how authorized consumers discover, contract, and receive event-based assets through secure and policy-controlled pipelines. Together, these workflows show how NSCA operationalizes Data Space principles within a scalable cloud architecture that supports cross-city collaboration, multi-stakeholder services, and long-term integration of smart-city intelligence.

Data provider perspective

On the provider side, an Event Platform emits new event data into the Data Bus – which implements a topic-based publish/subscribe mechanism. The Raw Event Sync component handles these events and triggers the synchronization operations required for updating the Event Repository in the Persistence layer, ensuring long-term storage of the event-based data. The Event/DS Explorer UI is a key component in the architecture that bridges the gap between the event data management and data space operations. It provides a user-friendly web interface that allows users to explore, configure, and manage the event-based data and their data sharing configurations. The Explorer acts as the main point of interaction with the Event Sync component, offering visibility into which data events are available, how they are structured and how they are exposed to the EDC Data Exchange component. The main function of the Event/DS Explorer UI is to enable filtering before event-based data assets are published to the data space. Users can select which subset of the events should be included in the shared data plane,

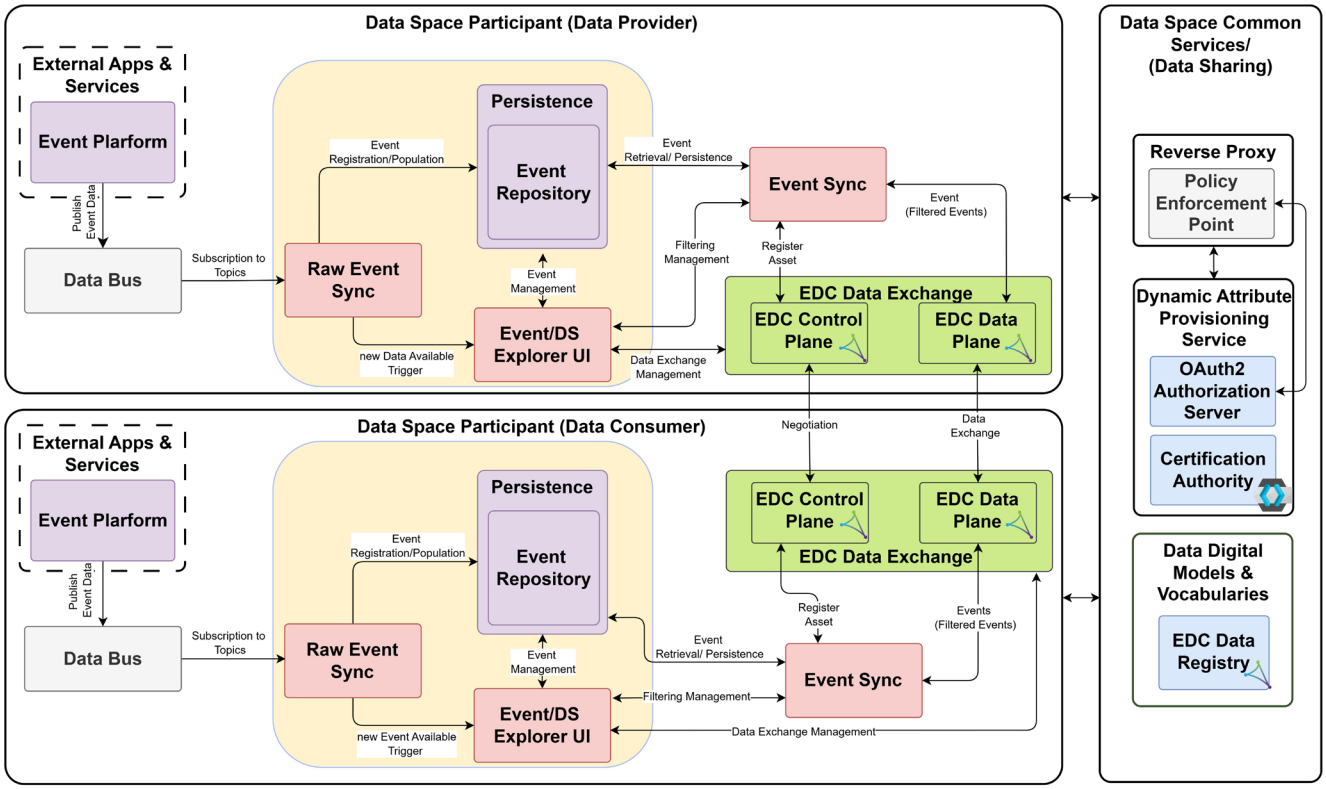

Figure 2. NCSA Data space architecture.

ensuring only relevant and approved information is made accessible to external parties. The Event Sync module registers the event-based data assets within the EDC Control Plane, where the corresponding metadata are published and made discoverable to data space participants. The EDC Data Plane then supports the secure and policy-compliant data transfer, delivering the filtered event-based data to authorized consumers.

Data consumer perspective

On the consumer side, a similar mechanism ensures seamless synchronization and consumption of event-based data assets. Through the Event/DS Explorer UI, consumers can discover available offerings, initiate data transfer workflows, and leverage the Event Sync component to receive and store event-based data assets to the Persistence layer based on specific configurations. The EDC Control Plane facilitates the discovery and negotiation of event-based data assets, while the EDC Data Plane executes the secure and policy-compliant transfer of data in accordance with the negotiated contracts and sharing policies established during the data exchange process.

Data space common services

The interactions between data space participants are governed and secured through the Data Space Common Services layer, which provides functionalities for the overall management, security, and interoperability of the dataspace. This layer incorporates the Policy Enforcement Point (PEP), which ensures that only authorized and authenticated users can access the platform's services, as well as the Dynamic Attribute Provisioning Service, supported by an OAuth2 Authorization Server and a Certification Authority. These components ensure that all data transactions comply with established identity, authentication, and trust policies among trusted participants. Furthermore, the EDC Data Registry, as part of the Data Digital Models and Vocabularies, maintains standardized metadata, digital asset descriptions, and interoperability vocabularies, thereby ensuring semantic consistency and discoverability of the exchanged information across the data space

Experimental evaluation

To demonstrate the expressiveness, flexibility, and generality of the Nous Smart City Architecture (NSCA), we present two representative services that showcase different interaction models, communication patterns, and deployment scenarios within the MASA testbed. These services collectively illustrate how SCA natively supports heterogeneous roles (assets, services, observers), semantic topic coordination, and distributed message flows across the edge–cloud continuum.

Vulnerable Road User (VRU) Service (Safety-Critical Asset Service)

The VRU Service is a real-time safety application designed to mitigate collision risks for pedestrians, cyclists, and other vulnerable road users (VRUs). It delivers personalized, low-latency safety alerts by fusing telemetry data from heterogeneous assets distributed across the environment. Interaction with the service is based on a registration model in which each participating asset subscribes to safety notifications relevant to its context.

Smart cameras and roadside units continuously publish object detections, including bounding boxes, object classes, and tracking identifiers, while mobile applications provide periodic GPS telemetry from user devices. For each registered asset, the VRU Service correlates visual perception from cameras with GNSS data from mobile devices to identify and track nearby entities in real time. Based on this correlated information, it evaluates proximity risk by considering relative distance, speed, and predicted trajectory, assigning each situation a severity level such as low, medium, or high. When a potential hazard is detected, the service generates a targeted alert and publishes it to dedicated notification channels so that only the relevant users and systems receive it. By enabling mission-critical, spatially aware reasoning across distributed and heterogeneous components, the VRU Service exemplifies the capabilities of the SCA architecture in real-world safety scenarios.

Beyond real-time alerting, the VRU Service also produces a continuous stream of structured, privacy-preserving metadata—such as aggregated detection events, risk-level distributions, near-miss indicators, and spatio-temporal mobility patterns—that can be exported to the NSCA Cloud layer and exposed through the Data Space. Once synchronized through the Event/DS Explorer and registered as data assets in the SIMPL/Open-compliant infrastructure, these datasets become securely discoverable and shareable with authorized third parties, including other municipalities, mobility agencies, road-safety authorities, and insurance operators. Because data sovereignty and usage policies are enforced at the connector level, providers can expose only high-level or anonymized aggregates while retaining full control over how the information is accessed, reused, or combined with external datasets. This enables cross-city benchmarking of safety hotspots, integration with regional or national mobility platforms, and the development of advanced risk-modelling services, all while ensuring compliance with GDPR and Data Space governance rules.

GeoPerception service (Edge Cooperative Awareness)

The GeoPerception Service enables localized cooperative perception by aggregating relevant environmental detections around a mobile user to enhance situational awareness. Unlike the alert-oriented VRU Service, its goal is not to infer risk but to reconstruct contextual understanding of the surrounding environment. It receives object detections from edge-deployed cameras, which continuously publish semantic labels, and GPS telemetry from mobile applications that may also specify query preferences, such as requesting only certain types of objects like vehicles. For every telemetry update received from a mobile asset, the service dynamically computes a spatial vicinity of 50 meters around the requester and filters incoming detections based on relevance, combining criteria such as distance, object class, and potential visibility constraints. The resulting localized perception view is then sent directly to the requesting asset through a dedicated notification channel. In doing so, the GeoPerception Service demonstrates how SCA supports dynamic edge-powered service composition, fine-grained spatial filtering, and low-latency analytics, all while significantly reducing bandwidth usage by transmitting only context-relevant information.

In addition to providing real-time, localized situational awareness to individual users, the GeoPerception Service produces a rich stream of structured environmental metadata—such as aggregated object counts, traffic density indicators, class distributions, and spatial perception snapshots—that can be synchronized to the cloud and exposed as Data Space assets. Once processed through the Event/DS Explorer and registered via SIMPL/Open connectors, these perception-based datasets become discoverable and consumable by authorized third parties, including mobility planners, transportation companies, digital-twin operators, and neighboring municipalities. Because the Data Space enforces strict usage policies and supports granular filtering, providers can share only anonymized, aggregated, or domain-specific subsets of the data, ensuring full compliance with GDPR and local governance requirements. This enables advanced applications such as inter-city mobility analytics, infrastructure planning, crowd-flow monitoring, and enhanced digital twins that combine perception data across multiple regions—ultimately fostering cross-city collaboration and more informed decision-making.

Experiments

To assess the latency and responsiveness of the NSCA communication infrastructure, we evaluated the end-to-end (E2E) delivery time of messages across the architecture. We first examined baseline communication latency under three representative deployment configurations, independent of any service logic, to isolate the overhead of NSCA's messaging fabric. We then analyzed data from two operational services deployed in MASA—VRU and GeoPerception—to evaluate NSCA under real-world producer–consumer interaction patterns. These measurements quantify the communication delays introduced by broker traversal, inter-broker forwarding, and optional message transformation, excluding any internal processing performed by services themselves. While VRU and GeoPerception are representative of safety-critical and perception services, the SCA architecture supports a broader set of urban applications with similar requirements.

The following architectural assumptions guide our evaluation:

1. Services are deployed at the Edge to model realistic latency-sensitive deployments.
2. Assets may be deployed either at the Edge or in the Cloud, depending on connectivity and mobility constraints.
3. Routing adapts dynamically to the location of message producers and consumers, using inter-broker forwarding only when necessary.

We evaluated E2E communication latency across three deployment pathways:

- *Cloud-to-Edge (CE):* a cloud-based asset publishes data to a cloud broker, which forwards it through the inter-broker bridge to an edge broker where the consuming service is located.
- *Edge-to-Cloud (EC):* an edge asset communicates with a cloud-based consumer through the edge broker and inter-broker bridge.
- *Edge-to-Edge (EE):* both producer and consumer operate within the same local edge domain, using only the edge broker.

These three paths correspond to the routing flows illustrated in Figure 1 (left). In the CE and EC cases, messages must traverse both brokers through the inter-broker bridge, adding network delay and security-layer overhead before reaching the destination. In contrast, EE communication remains confined to the local edge broker, resulting in the lowest and most predictable latency. This distinction highlights how NSCA's hybrid broker deployment enables flexible routing: local interactions are kept within the edge domain for real-time processing, while supervision, cloud-level consumers, and cross-domain services can still access the necessary data through controlled inter-broker forwarding.

Figure 3 decomposes these latencies into three propagation segments: (i) source asset → source broker, (ii) inter-broker forwarding, and (iii) destination broker → destination asset. As expected, the Edge-to-Edge path delivers the lowest latency due to local routing without inter-broker traversal making it ideal for real-time services such as VRU safety alerts. In both CE and EC scenarios, delays arise from WAN propagation and security overhead at cloud ingress. The EC path is slightly faster than CE due to asymmetric ingress filtering and NAT/firewall rules applied to incoming traffic at the edge perimeter.

In another experiment we test the Impact of Message and Object Filtering. The GeoPerception service performs spatial filtering of incoming perception data to reduce bandwidth consumption and computation for downstream consumers. To measure this effect, we monitored the cumulative data received by a client over time with and without filtering. As shown in Figure 4. We created a linear regression to model the payload/objects being sent over time, and compare the slope to understand the rate of savings.

- Payload volume increases with a slope of 12.7 KB/s without filtering and with 2.7 KB/s with filtering.
- The number of detected objects delivered to the consumer increases with a slope of 37.7 objects/s without filtering and with 4.9 objects/s with filtering.

The slightly lower byte reduction compared to object reduction is due to message headers and metadata that must be preserved, even when no objects are present. These results show that moving filtering and aggregation to the Edge significantly reduces network and processing overhead, preventing unnecessary traffic from entering the Cloud or burdening downstream applications. This confirms the effectiveness of SCA's broker-level filtering strategy for scalable perception services.

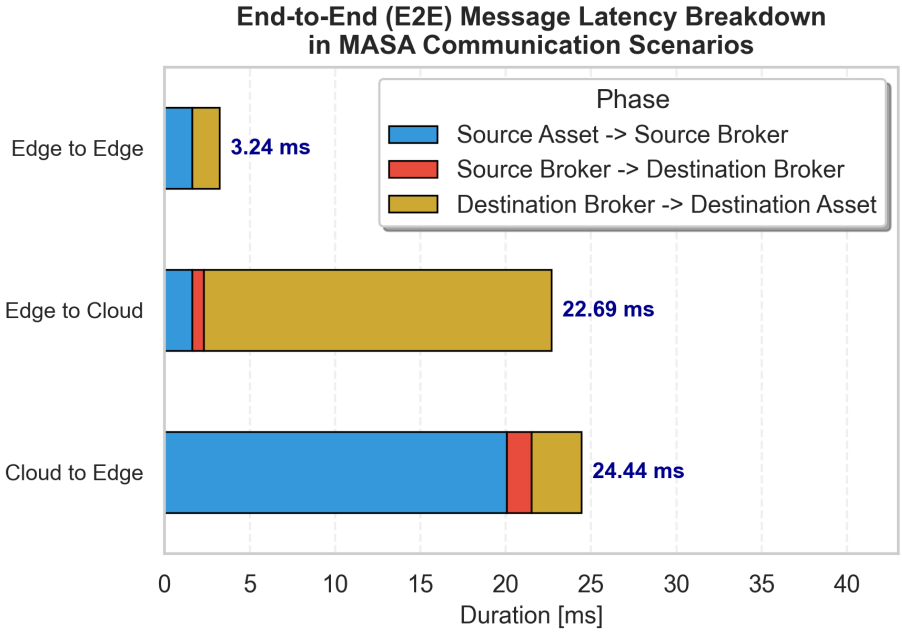

Figure 3. E2E message delivery latency across three representative deployment scenarios decomposed into three phases.

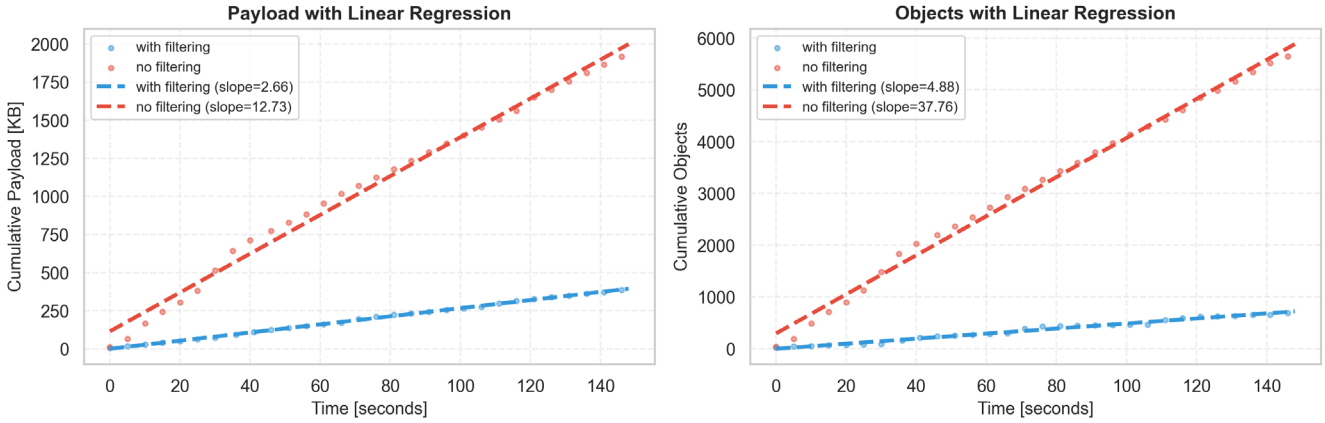

Figure 4. Cumulative data received with and without GeoPerception filtering. Blue lines represent the filtered messages delivered to the consumer; green lines represent the unfiltered input received by the GeoPerception service. Left: Total payload (Kbytes) over time. Right: Total number of detected objects. The coefficient of the linear regression highlight the rate of saving using filtering.

Finally, To evaluate interaction responsiveness, we analyzed the full message exchange timeline of the VRU service, from broker connection to service unsubscription. The interaction comprises the following phases:

1. Connection to broker
2. Service discovery
3. Service registration
4. First telemetry request-response cycle
5. Second telemetry request-response cycle
6. Service unsubscription

The first telemetry response includes initialization overhead, while subsequent interactions show steady low latency due to session reuse. We tested four deployment scenarios, comparing Cloud vs Edge connectivity and 5G vs ethernet connectivity. Results (Figure 5) show that:

- Ethernet at the Edge delivers the best performance, achieving minimal and stable latency.

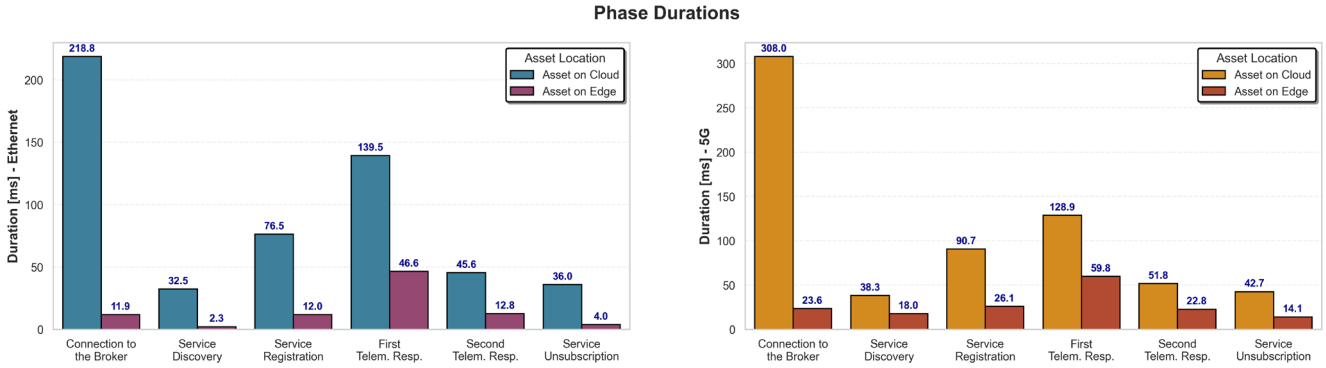

Figure 5. End-to-end interaction timeline of the VRU service across different connectivity scenarios. Left: Asset located at the Edge with Ethernet. Right: Same deployment over 5G NSA, introducing consistent latency overhead.

- 5G NSA via MEC introduces an extra 10–15 ms per interaction phase, but remains suitable for mobile VRU deployments thanks to consistent latency.
- Cloud-based assets show the highest and most unstable latency, due to WAN traversal, inter-broker routing, and public internet congestion.
- Unsubscription is consistently lightweight, demonstrating the efficiency of topic-based interaction coordination.

These findings emphasize that edge deployment is essential for time-critical safety services, while 5G MEC offers a practical compromise between mobility and latency. Cloud-only deployments are functional but unsuitable for strict real-time guarantees.

Although these experiments primarily assess the performance of the NSCA messaging layer for real-time edge services, their implications extend directly to the Data Space integration in the cloud. Efficient edge-to-cloud routing, predictable latency, and aggressive broker-level filtering directly influence how event-based datasets are synchronized, catalogued, and exposed as data assets in the SIMPL/Open ecosystem. Lower propagation delays ensure that aggregated mobility and perception data reach the cloud layer in near–real time, enabling timely updates of shared assets, digital twins, and analytics pipelines. Similarly, reducing unnecessary payloads at the edge minimizes cloud ingestion overhead, improves the scalability of the Event Sync component, and ensures that only meaningful, policy-compliant information enters the Data Space. As a result, the performance characteristics demonstrated in these experiments are foundational not only for real-time applications like VRU and GeoPerception, but also for the reliability, cost-efficiency, and responsiveness of multi-stakeholder data exchange within the NOUS Data Space.

Related work

Recent advances in smart city infrastructures have underscored the need for scalable and adaptive communication architectures capable of handling heterogeneous data streams, latency-sensitive services, and distributed intelligence across the IoT–Edge–Cloud continuum. Conventional centralized platforms—typically built around large-scale data warehouses or distributed stream-processing frameworks such as Apache Kafka and Apache Flink—provide strong support for high-throughput analytics[5,6]. However, their rigid topologies, tightly coupled data pipelines, and complex deployment lifecycles make them less suitable for dynamic, event-driven environments where responsiveness and modularity at the network edge are essential.

To address these architectural constraints, the Data Mesh paradigm has emerged as a decentralized alternative, advocating domain-oriented data ownership, self-serve data infrastructure, and federated governance to promote interoperability at scale[7]. While promising, Data Mesh approaches often overlook the real-time communication and synchronization requirements typical of cyber-physical urban systems.

This paper builds upon our previous research in edge computing, cooperative safety services for Vulnerable Road Users (VRUs), and distributed service orchestration, introducing a Smart City Architecture (SCA) that combines broker-level extensibility, topic-driven interaction semantics, and publish/subscribe service coordination. The design is inspired by earlier contributions that explored 5G MEC-enabled low-latency edge-to-cloud communication and adaptive orchestration mechanisms for distributed services. In particular, VRU collision risk mitigation using MEC infrastructures was evaluated in 8, while broker-side computation and semantic message routing were enabled through a plugin-based MQTT extension introduced in 1. Complementary work demonstrated urban sensor fusion for VRU risk prediction and investigated dynamic orchestration across heterogeneous computing tiers[9–13]. Additionally, real-world deployments of smart city platforms and data mesh-inspired designs in urban contexts were reported in 1,14. The SCA framework consolidates these research directions by providing a unified, MQTT-native middleware with modular client registration and multi-role entity interaction.

Within the mobility and safety domain, related efforts include V2I/V2P services for proactive VRU protection[15,16], machine learning-based pedestrian intent prediction[17], and augmented reality interfaces for cooperative awareness[18]. Further advancements in ADAS systems leveraging 5G and edge computing were discussed in 19,20. SCA extends this line of work by enabling fine-grained messaging at the client level instead of relying on generic broadcast-based patterns, improving privacy, scalability, and selective information dissemination through topic-based registration and edge-based message routing.

With regard to MQTT-based distributed systems, several contributions have explored federated broker overlays[21,22] and hierarchical routing strategies for publish/subscribe networks[23,24], though often at the expense of architectural simplicity or maintainability. Alternative efforts have proposed MQTT gateways with message filtering and AI-based traffic prioritization[25,26], but these approaches typically introduce additional operational layers and deployment overhead. In contrast, SCA adopts a broker-native approach that preserves the simplicity of MQTT while enabling runtime extensibility via plugins, supporting YAML-configurable message transformations, semantic routing rules, and minimal processing latency across distributed edge deployments.

Recent research on European Data Spaces and federated data-sharing infrastructures has increasingly emphasized the need for trusted, interoperable, and sovereignty-preserving mechanisms for cross-organizational data exchange. The International Data Spaces (IDS) architecture has been one of the earliest and most influential frameworks in this area, introducing certified connectors, usage-control policies, and standardized contract negotiation to ensure secure, sovereign data sharing among independent organizations[27]. Building on IDS principles, the GAIA-X initiative proposes a federated cloud and data ecosystem where participants interconnect through common services for identity, trust, and catalog federation, with strong alignment to European regulations on privacy, security, and transparency[28]. These foundations have directly informed the design of sector-specific European Data Spaces, such as the European Mobility Data Space (EMDS) and the Smart Communities Data Space, which promote cross-city interoperability and semantically harmonized data models across public administrations and mobility service providers.

Within this broader ecosystem, the European Commission recently introduced SIMPL/Open (Smart Middleware for Trusted Data Spaces) as a unifying middleware architecture that operationalizes Data Space concepts with modular, open-source building blocks for connectors, policy enforcement, semantic registries, and identity services[4,29]. SIMPL/Open aims to reduce fragmentation by providing a reference implementation aligned with the upcoming Data Act, Data Governance Act, and High-Value Data frameworks, ensuring that public administrations and private stakeholders can deploy interoperable Data Space solutions with minimal integration overhead. Recent technical reports highlight the importance of modularity, extensibility, and common APIs in enabling cities, industrial clusters, and research institutions to participate in federated data ecosystems without losing control over their assets.

Further academic work has examined the challenges of semantic interoperability and cross-domain data governance in Data Spaces, with several studies exploring ontology alignment, metadata standardization, and policy-compliant data processing pipelines for multi-stakeholder environments[2,30]. These contributions complement SIMPL/Open by emphasizing the need for shared vocabularies, machine-readable governance rules, and automated compliance checking—capabilities that are crucial for smart city deployments where datasets originate from heterogeneous cyber-physical systems. Together, these initiatives position Data Spaces as a core enabler for next-generation smart city platforms and provide the conceptual and operational building blocks that NOUS integrates within its cloud architecture.

Conclusion

This paper presented the Nous Smart City Architecture (NSCA), a modular and extensible middleware that unifies real-time edge intelligence with sovereign and interoperable data sharing in the cloud. Validated within the Modena Automotive Smart Area (MASA), NSCA demonstrates how lightweight publish/subscribe communication, semantic topic structuring, and federated brokers can support low-latency coordination among heterogeneous urban assets at the network edge. At the same time, its cloud-side integration with SIMPL/Open enables event synchronization, metadata curation, and policy-controlled publication of urban data assets within a Data Space ecosystem. Together, these two layers provide a seamless continuum—from millisecond-level edge processing to multi-stakeholder cloud interoperability.

Experimental results confirmed NSCA's effectiveness in latency-sensitive and bandwidth-constrained scenarios, showing how edge-local message routing, spatial filtering, and plugin-based transformations significantly reduce communication overhead and improve responsiveness. The deployment of the VRU and GeoPerception services further demonstrated how NSCA supports both real-time safety applications and cloud-level data enrichment, enabling their outputs to be shared across municipalities, operators, and third-party analytics platforms through the Data Space.

Future work will expand NSCA along both dimensions. On the edge side, we plan to enhance multi-protocol interoperability (AMQP, DDS, NGSI-LD), strengthen QoS guarantees, and extend plugin-based processing with edge analytics and anomaly detection. On the Data Space side, we aim to broaden semantic interoperability, automate usage-policy enforcement, and improve cross-city federation through standardized connectors and governance models aligned with emerging European initiatives for mobility and smart communities.

By advancing both the edge middleware and the cloud Data Space infrastructure, NSCA aspires to become a comprehensive foundation for next-generation smart city platforms—one that supports real-time intelligence, secure data exchange, and large-scale collaboration across diverse urban stakeholders.

Ethics and consent

Ethical approval and consent were not required

Data availability

This paper describes software architecture, so there is no data associated with this publication.

Software availability

Source code available from: https://gitlab.eclipse.org/eclipse-research-labs/nous-project

Archived software available from: https://doi.org/10.5281/zenodo.17972799

References

1. Pietri M, Taccini L, Bedogni L, *et al.*: **Efficient and flexibile IoT communication through a plugin-based MQTT processing architecture**. In: *International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*. 2025.
Publisher Full Text
2. Otto B: **A federated infrastructure for European data spaces.** *Commun ACM.* 2022; **65**(4): 44–45.
Publisher Full Text
3. Scerri S, Tuikka T, de Vallejo IL, *et al.*: **Common European data spaces: challenges and opportunities**. *Data Spaces: Design, Deployment and Future Directions*. 2022; 337–357.
Publisher Full Text
4. European Commission: **Smart middleware for trusted data spaces (simpl)**. 2024.
Reference Source
5. Machado IA, Costa C, Santos MY: **Data mesh: concepts and principles of a paradigm shift in data architectures.** *Procedia Comput Sci.* 2022; **196**: 263–271.
Publisher Full Text
6. Žaja M, Čavrak I, Lipić T: **Benchmarking apache beam for IoT applications**. In: *International Convention on Information, Communication and Electronic Technology*. 2021.
Publisher Full Text
7. Goedegebuure A, Kumara I, Driessen S, *et al.*: **Data mesh: a systematic gray literature review.** *ACM Comput Surv.* 2024; **57**(1): 1–36, 11.
Publisher Full Text
8. Rossini E, Pietri M, Picone M, *et al.*: **Vulnerable Road Users accident prevention via smart city data fusion: experimental evaluation of a 5G MEC architecture**. In: *International Symposium on Network Computing and Applications*. 2024.
Publisher Full Text
9. Bedogni L, Picone M, Pietri M, *et al.*: **Fluid computing in the internet of things: a digital twin approach.** In: *IEEE Consumer Communications & Networking Conference (CCNC)*. 2024.
Publisher Full Text
10. Bedogni L, Mamei M, Picone M, *et al.*: **Fluid computing & digital twins for intelligent interoperability in the IOT ecosystem.** *Future Gener Comput Syst.* 2025; **171**: 107855.
Publisher Full Text
11. Picone M, Bedogni L, Pietri M, *et al.*: **Digital twins & fluid computing in the edge-to-cloud compute continuum**. In: *IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*. 2024.
Publisher Full Text
12. Picone M, Bedogni L, Pietri M, *et al.*: **Dynamic function validation and simulation in fluid digital twins**. In: *International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*. 2024.
Publisher Full Text
13. Picone M, Barbone A, Morandi R, *et al.*: **From physical to digital: exploring digital twins within the Modena Automotive Smart Area**. In: *IEEE Consumer Communications Networking Conference*. 2025.
Publisher Full Text
14. Rossini E, Bicocchi N, Hadjidimitriou NS, *et al.*: **Towards a distributed data mesh model for the IoT-edge-cloud continuum in smart cities**. In: *IEEE/ACM Symposium on Edge Computing*. 2024.
Publisher Full Text
15. Lusvarghi L, Grazia CA, Klapez M, *et al.*: **Awareness messages by Vulnerable Road Users and vehicles: field tests via lte-v2x.** *IEEE Transactions on Intelligent Vehicles.* 2023; **8**(10): 4418–4433.
Publisher Full Text
16. Papadimitratos P, De La Fortelle A, Evenssen K, *et al.*: **Vehicular communication systems: enabling technologies, applications, and future outlook on intelligent transportation.** *IEEE Commun Mag.* 2009; **47**(11): 84–95.
Publisher Full Text
17. Rasouli A, Kotseruba I, Tsotsos JK: **Understanding pedestrian behavior in complex traffic scenes.** *IEEE Transactions on Intelligent Vehicles.* 2018; **3**(1): 61–70.
Publisher Full Text
18. Berge SH, de Winter J, Hagenzieker M: **Support systems for cyclists in automated traffic: a review and future outlook.** *Appl Ergon.* 2023; **111**: 104043.
PubMed Abstract | **Publisher Full Text**
19. Feng L, Li W, Lin Y, *et al.*: **Joint computation offloading and URLLC resource allocation for collaborative MEC assisted cellular-V2X networks.** *IEEE Access.* 2020; **8**: 24914–24926.
Publisher Full Text
20. Yang YS, Lee SH, Chen GS, *et al.*: **An implementation of high efficient smart street light management system for smart city.** *IEEE Access.* 2020; **8**: 38568–38585.
Publisher Full Text
21. Longo E, Redondi AEC, Cesana M, *et al.*: **MQTT-ST: a spanning tree protocol for distributed MQTT brokers.** In: *IEEE International Conference on Communications (ICC)*. 2020.
Publisher Full Text
22. Stagliano L, Longo E, Redondi AEC: **D-MQTT: design and implementation of a pub/sub broker for distributed environments**. In: *IEEE Int Conf Omnilayer Intell Syst*. 2021.
Publisher Full Text
23. Chockler G, Melamed R, Tock Y, *et al.*: **Constructing scalable overlays for pub-sub with many topics.** In: *ACM Symposium on Principles of Distributed Computing.* 2007; 109–118.
Publisher Full Text
24. Fidler E, Jacobsen HA, Li G, *et al.*: **The PADRES distributed publish/subscribe system.** In: *Principles and applications of distributed event-based systems.* 2010.
Publisher Full Text
25. Beniwal G, Singhrova A: **A systematic literature review on IOT gateways** *J King Saud Univ Comput Inf Sci.* 2022; **34**(10): 9541–9563.
Publisher Full Text
26. Toskov B, Toskova A, Bogdanov S, *et al.*: **Intelligent IoT gateway**. In: *International Conference Automatics and Informatics*. 2021.
Publisher Full Text
27. Usländer T, Teuscher A: **Industrial data spaces**. In: *Designing data spaces*. 2022.
Publisher Full Text
28. GAIA-X AISBL: **GAIA-X architecture document**.
Reference Source
29. European Commission: **European mobility data space - vision paper.** 2023.
Publisher Full Text
30. Fensel D, Şimşek U, Angele K, *et al.*: **Knowledge graphs: methodology, tools, and selected use cases.** Springer, 2021.
Publisher Full Text

Open Peer Review

Current Peer Review Status: ? ?

Version 1

Reviewer Report 22 January 2026

https://doi.org/10.21956/openreseurope.24022.r67714

?

Eirini Eleni Tsiropoulou
Arizona State University, Tempe, Arizona, USA

The paper introduces the NOUS Smart City Architecture (NSCA), a comprehensive middleware framework designed to bridge the edge-cloud continuum for intelligent urban services.

Though the manuscript is overall well written and easy to follow there are several aspects that they need improvement and further clarification the revised version of the manuscript.

What is the specific mechanism for configuring this selection (e.g., a YAML file, a dynamic API)?

How is topic namespace collision avoided when multiple independent edge sites (e.g., different city districts) forward topics to the same cloud broker?

Your broker plugin is a powerful feature for edge-side filtering. Can you describe its internal architecture in more detail? Specifically, does it maintain any state (e.g., for spatial filtering across messages), and if so, how is this state managed in a multi-threaded broker environment?

A more thorough discussion of the current state of the art is needed especially dealing with the uncertainty within the computing environment. Please elaborate more on the risk-aware data offloading in multi-server multi-access edge computing environment.

What is the performance overhead you measured for a plugin performing a spatial filter compared to a simple topic-based subscription?

The Data Space integration relies on an Event Sync component. Could you detail its internal workflow?

Does it consume messages from the cloud broker, buffer them, batch them, and then push them to the SIMPL/Open EDC?

How does it handle message ordering guarantees and delivery assurances (at-least-once, exactly-

once) between the MQTT world and the Data Space world?

It would be great if the authors can address those comments in the revised version of the manuscript.

Is the work clearly and accurately presented and does it cite the current literature?
Yes

Is the study design appropriate and does the work have academic merit?
Yes

Are sufficient details of methods and analysis provided to allow replication by others?
Yes

If applicable, is the statistical analysis and its interpretation appropriate?
Yes

Are all the source data underlying the results available to ensure full reproducibility?
Yes

Are the conclusions drawn adequately supported by the results?
Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: reinforcement learning

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Reviewer Report 16 January 2026

https://doi.org/10.21956/openreseurope.24022.r67717

?

Shiva Shankar Reddy
Sagi Rama Krishnam Raju Engineering College (A), Bhimavaram, Andhra Pradesh, India

1. See that the abstract section contains with the introduction, data used, objectives, models used, results, and conclusion.
2. The content was written in a general way. Need to cite the data with the latest articles. See that the paragraphs should have the connectivity in between them.

3. The subheadings of Figure 1 and 2 need a clear explanation of each.
4. Explain how far the NSCA supports multi-role participation by allowing a single application to assume multiple roles concurrently.
5. How the interaction between assets and services is the observer role.
6. How far the Dynamic plugin and filtering enhance flexibility and control over message handling within SCA.
7. Explain how NSCA empowers to incorporate cooperative edge intelligence without modifying client applications or services.
8. Explain how the NOUS Data Space acts as the cloud-side extension of NSCA, how it enables real-time data, and how to create value-added services.
9. Explain how the interactions between data space participants are secured through the Data Space Common Services layer.
10. A brief explanation is needed for representative services and how they differ in interaction models, communication patterns, and deployment scenarios within the MASA testbed.
11. How the writer evaluates proximity risk by considering relative distance, speed, and predicted trajectory, assigning each situation a severity level.
12. Explain how the GeoPerception Service enables the localized cooperative perception by aggregating relevant environmental detections around a mobile user to enhance situational awareness.
13. Explain how figures 3, 4, and 5 were generated. How the values were obtained for the tables.
14. The conclusion section should be precise and should be covered with the results obtained.
15. The discussion section should be added by citing the latest articles. Need to prove that the proposed approaches were better compared with the existing ones.

Is the work clearly and accurately presented and does it cite the current literature?
No

Is the study design appropriate and does the work have academic merit?
Partly

Are sufficient details of methods and analysis provided to allow replication by others?
Partly

If applicable, is the statistical analysis and its interpretation appropriate?
Partly

Are all the source data underlying the results available to ensure full reproducibility?
Partly

Are the conclusions drawn adequately supported by the results?
No

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Data Mining, Medical Mining, Machine Learning, Deep Learning, Edge Compuitng, Cloud computing, Image Processing and IoT

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.