Architecting
Digital Twins for Intelligent Transportation Systems

Hiya Bhatt*, Sahil†, Karthik Vaidhyanathan*, Rahul Biju†, Deepak Gangadharan†, Ramona Trestian‡, Purav Shah‡

*Software Engineering Research Centre, IIIT Hyderabad, India {hiya.bhatt@research.iiit.ac.in, karthik.vaidhyanathan@iiit.ac.in}
†Computer Systems Group, IIIT Hyderabad, India {sahil.sahil@research.iiit.ac.in, rahul.biju@research.iiit.ac.in, deepak.g@iiit.ac.in}
‡Faculty of Science and Technology, Middlesex University London {r.trestian@mdx.ac.uk, p.shah@mdx.ac.uk}

Abstract—Modern transportation systems face growing challenges in managing traffic flow, ensuring safety, and maintaining operational efficiency amid dynamic traffic patterns. Addressing these challenges requires intelligent solutions capable of real-time monitoring, predictive analytics, and adaptive control. This paper proposes an architecture for DigIT, a Digital Twin (DT) platform for Intelligent Transportation Systems (ITS), designed to overcome the limitations of existing frameworks by offering a modular and scalable solution for traffic management. Built on a Domain Concept Model (DCM), the architecture systematically models key ITS components enabling seamless integration of predictive modeling and simulations. The architecture leverages machine learning models to forecast traffic patterns based on historical and real-time data. To adapt to evolving traffic patterns, the architecture incorporates adaptive Machine Learning Operations (MLOps), automating the deployment and lifecycle management of predictive models. Evaluation results highlight the effectiveness of the architecture in delivering accurate predictions and computational efficiency.

Index Terms—Digital Twin, Intelligent Transportation System, Domain Driven Design, MLOps

I. INTRODUCTION

The increasing complexity of modern transportation systems, characterized by growing vehicular density, diverse communication protocols, dynamic and distinct traffic patterns, and complex road networks poses significant challenges in traffic management, safety, and operational efficiency. To this end, Intelligent Transport System (ITS) has emerged as a potential solution as they serve as an ecosystem that integrates technology, communication, and data analytics to improve the efficiency, safety, and sustainability of transportation systems. ITS solutions are used to manage and optimize traffic flow, enhance user experience, and support advanced applications like autonomous driving and vehicular communication [1], [2]. Most importantly, the ITS infrastructure helps reduce the accidents that occur on the roads. Hence, it is important to deploy more ITS infrastructure as fatal accidents are on the rise in several countries [3], [4].

However, deploying an ITS infrastructure, which achieves the earlier mentioned objectives while ensuring safety, is non trivial due to its scale, dynamic environment, and real-time demands. The development of a formally sound ITS solution thus becomes challenging as it is difficult to verify all the what-if scenarios in the physical deployment. In response, Digital Twins (DTs) have emerged as a transformative paradigm, enabling the creation of virtual replicas of physical systems to support real-time monitoring, predictive analytics, and adaptive control mechanisms [5] [6] [7] [8]. In the context of ITS, DTs leverage real-time data from IoT-enabled sensors, vehicular networks, and environmental monitoring systems to improve situational awareness and enhance decision-making processes [9]. Existing implementations, such as Mobility DT [10], have demonstrated their utility in modeling traffic flows, evaluating rerouting strategies, and predicting congestion patterns through simulation-based approaches. These advancements highlight the potential of DTs to transform urban mobility and transportation planning.

Although modeling traffic flows works well in several countries, the challenge is significantly higher in many other countries that witness highly unstructured traffic. This scenario also demands robust traffic management solutions to mitigate the social, economic, and environmental impacts of congestion. Traffic flow prediction enables proactive decision-making and efficient traffic control. However, the irregular and dynamic nature of traffic patterns, presents unique challenges in identifying consistent trends. To address these complexities, this project leverages temporal and spatio-temporal Deep Learning (DL) models, such as LSTM (Long Short-Term Memory) [11] and BiLSTM (Bidirectional LSTM) [12]. LSTM, a variant of Recurrent Neural Networks (RNNs), excels in capturing long-term dependencies in sequential data, making it highly effective for modeling traffic patterns that evolve over time. Its ability to retain and utilize information over long sequences enables accurate predictions of future traffic flows based on historical trends. By integrating real-time data from traffic nodes at intersections in Hyderabad, India, the system ensures precise predictions and real-time adaptability. Automated retraining mechanisms further enhance robustness, contributing to smarter and more efficient traffic management systems.

This paper introduces architecture for $\mathcal{D}igIT$ (DT for ITS) platform, addressing the limitations of existing frameworks by providing a modular, scalable, and adaptive solution for traffic management. To systematically model the components and behaviors of ITS, the architecture adopts a *Domain Concept Model (DCM)*, which captures key entities such as vehicles, sensors, communication networks, and user behaviors. The DCM serves as the foundation for designing the architecture and defining the interactions required to simulate traffic scenarios and predict outcomes effectively.

The proposed architecture integrates real-time data streams with Machine Learning (ML) based predictive modeling to optimize traffic flow and enhance decision-making. It employs ML models to forecast traffic patterns based on historical and real-time data. The predictions are validated through simulations, enabling scenario testing and evaluation of intervention strategies. By bridging modeling and simulation, this architecture demonstrates the potential of integrating predictive analytics into real-

time traffic management workflows. The architecture for $\mathcal{D}igIT$ platform incorporates adaptive *Machine Learning Operations (MLOps)* [13] practices, automating the deployment and lifecycle management of predictive models. This framework supports dynamic adjustments to evolving traffic conditions while maintaining computational efficiency and scalability. The results presented in this work validate the system's accuracy and responsiveness, demonstrating its suitability for real-world ITS applications.

II. RELATED WORK

Digital Twin (DT) technology has shown significant potential across diverse domains by enabling real-time monitoring, predictive analytics, and scenario-based simulations. Its adaptability has made it particularly effective in applications ranging from infrastructure management to smart cities, offering a foundation for addressing the challenges of Intelligent Transportation Systems (ITS).

In infrastructure management, Likhit et al. [14] demonstrated DTs' effectiveness in optimizing water distribution networks through real-time data and simulations. Their work highlights leveraging predictable flow patterns to improve resource management. However, transportation systems add complexity due to unpredictable traffic dynamics and human behavior, as noted by Glaessgen and Stargel [15], requiring robust frameworks to address uncertainties.

Expanding to urban-scale applications, Mohammadi et al. [16] applied DTs for smart city optimization, enabling better decision-making and resource allocation. Similarly, Barat et al. [17] modeled pandemic dynamics, emphasizing adaptability in crisis scenarios. Despite these advances, Barn et al. [18] highlighted gaps in integrating socio-technical interactions and behavioral modeling, limiting realism in existing frameworks.

To address scalability and interoperability, Redelinghuys et al. [19] proposed layered DT architectures, promoting modularity. Complementing this, Kulkarni et al. [20] introduced automated workflows for predictive modeling, ensuring responsiveness under evolving conditions. Yet, these methods often rely on static configurations, posing challenges for dynamic systems requiring continuous retraining.

Despite advancements, Digital Twin (DT) systems still struggle with real-time synchronization between virtual and physical environments, often relying on offline simulations [21]. Recent efforts, such as Ge et al. [22], introduced cyber-physical frameworks with IoT sensors for synchronized traffic simulations. However, scalability to multi-modal urban systems remains a challenge [23].

To tackle these limitations, Domain-Driven Design (DDD) offers a modular approach for structuring complex DTs. By leveraging bounded contexts, DDD improves scalability and adaptability. Macias et al. [24] applied DDD principles for scalable architectures, Evans [25] emphasized adaptability and automated updates, highlighting DDD's potential to meet real-time demands and evolving requirements.

Building on this foundation, we propose an architecture for the $\mathcal{D}igIT$ platform—a DT framework for ITS that integrates predictive modeling, simulations, and automated workflows. The architecture applies DDD principles to identify domains, which are modeled using a Decision-Component Model (DCM). The DCM captures key entities such as vehicles, sensors, communication networks, and user behaviors, ensuring seamless integration between physical and digital systems. ML models handle traffic forecasting, supported by MLOps pipelines for scalability through automated retraining. Simulations enable scenario testing, providing insights for decision-making and interventions. By combining these elements, our framework addresses modern transportation complexities with a dynamic and scalable approach.

III. DEPLOYED ITS USE CASE

The development of a DT for ITS involves deploying IoT-based traffic monitoring systems in urban areas to create a comprehensive and scalable platform for understanding and optimizing traffic flow. The focus is on real-time data collection, simulation of traffic scenarios, and predictive analysis to address challenges such as congestion, delays, and inefficiencies in traffic management.

To develop an exemplar ITS system, a network of edge devices are strategically deployed at busy intersections near the IIIT Hyderabad campus. These devices are equipped with cameras and processors that capture real-time traffic data, including vehicle counts, traffic density, and congestion levels, at regular intervals. The data is processed locally on the edge devices using advanced temporal models such as LSTM (Long Short-Term Memory) and BiLSTM (Bidirectional LSTM), which are highly effective in capturing traffic flow patterns and predicting future trends. By leveraging LSTM's ability to model long-term dependencies and BiLSTM's bidirectional analysis of sequential data, the system ensures accurate and timely predictions.

The processed data is aggregated into 5-minute intervals to reduce noise and optimize storage and is then transmitted to a public API hosted on a Virtual Private Server (VPS). This API serves as a centralized hub, enabling other system components to access the data seamlessly. Additionally, the edge devices' local processing minimizes latency, ensuring quick responses to real-time traffic conditions.

This setup provides continuous insights into traffic flow dynamics, particularly during peak hours or in response to unexpected disruptions like accidents or roadblocks. The integration of LSTM and BiLSTM models enhances the system's ability to adapt to the unpredictable and highly dynamic nature of traffic patterns. Furthermore, the system supports regular retraining mechanisms, either scheduled or dynamically triggered by performance thresholds, to ensure the predictions remain accurate and relevant. This comprehensive approach forms the backbone of real-time traffic management and simulation systems, enabling more efficient and adaptive decision-making processes. The stored data powers simulations using tools like the Simulation of Urban MObility (SUMO) [26], which replicate real-world traffic conditions to model the behavior of vehicles across the monitored intersections. For example, simulations might explore how adjusting signal timings or rerouting traffic can mitigate bottlenecks during heavy congestion. Alongside simulations, ML models analyze both real-time and historical data to forecast future traffic conditions, such as predicting increased congestion at specific intersections during evening rush hours.

This use case demonstrates how DT can enable a better understanding of traffic flow in urban environments. By combining real-time data collection, simulation, and predictive analytics, the system offers tools for traffic managers to monitor current conditions and test interventions virtually before

implementing them. We utilize this use case in the remainder of this paper to contextualize and elaborate on the proposed architecture for building and deploying a DT for ITS.

IV. ARCHITECTURE FOR DIGITAL TWIN

The design and implementation of the architecture of $\mathcal{D}igIT$ (DT for ITS) platform require a structured approach to address the inherent complexities of urban mobility. Using the Sociotechnical DT Specification Meta Language Concepts [18] as the foundation, this approach establishes the scope, purpose, and operational mechanisms of the DT. At the core of this system is the Domain Concept Model (DCM), which provides a structured representation of the architecture of $\mathcal{D}igIT$ components and their interactions. Together, these elements ensure that the architecture for $\mathcal{D}igIT$ platform is both theoretically sound and practically viable.

A. Sociotechnical DT Specification Meta Language Concepts for DigIT

The specification meta-language serves as a framework to articulate and guide the development of the DT for ITS. Each concept within this framework is defined and applied explicitly to address the challenges and requirements of ITS.

Goal. The primary goal of the DT is to enhance traffic management by enabling proactive and data-driven decision-making. Key objectives include improving traffic flow efficiency, reducing congestion and delays, and enhancing road safety. Additionally, the DT seeks to expand and optimize communication capabilities through adaptive technologies, ensuring seamless information exchange between vehicles and infrastructure. These goals collectively aim to create a more responsive and sustainable ITS.

Measure. The effectiveness of the DT is evaluated using quantifiable metrics that assess traffic performance, environmental impact, and communication reliability. These measures include average travel time per vehicle, intersection throughput (vehicles per minute), reduction in greenhouse gas emissions, and the number of accidents or incidents reported per month. Additional metrics, such as average vehicle speed during peak hours and communication latency between traffic management systems and vehicles, provide insights into system responsiveness and efficiency. These indicators ensure the DT delivers measurable improvements aligned with ITS objectives.

Assumption. The assumptions explain the scope of the DT and define the prerequisites for its successful deployment. For the ITS-focused DT, the architecture assumes the presence of IoT-enabled sensors deployed across the transportation network. These sensors capture real-time data, such as traffic flow rates, vehicular speeds, and congestion levels. Environmental sensors are also assumed to provide supplementary data on external factors like weather conditions. Additionally, the architecture assumes the existence of a robust communication infrastructure that facilitates the seamless transmission of data between the physical and digital layers. These assumptions ensure that the DT operates in a data-rich environment, accurately reflecting real-world conditions [27].

Lever/Policy. Levers or policies are quantifiable or actionable elements that influence the operation of the DT. Examples of policies include dynamic traffic signal adjustments, traffic rerouting strategies, emission control policies, integration with public transit policies, pedestrian prioritization, and enabling direct vehicle-to-vehicle (V2V) communication for dynamic traffic information sharing. These levers serve as input for simulations and predictive models, allowing the DT to anticipate traffic conditions and recommend appropriate interventions. For example, a high vehicle density at a particular intersection could be detected and processed by an ML model, which predicts the potential impact of congestion. This prediction could then initiate simulations to assess the effectiveness of rerouting strategies or signal adjustments to alleviate the congestion.

Regulation. Regulations define enforceable policies and standards that guide the operations of the DT within the ITS. These include city-specific traffic and parking rules, privacy laws governing data collection and use, emission control limits for urban zones, and interoperability standards for IoT and ITS components. Such regulations ensure that the DT complies with societal, environmental, and legal requirements. For example, emission control policies can trigger restrictions on vehicle performance during periods of high pollution, while interoperability standards ensure seamless integration between heterogeneous systems within the ITS ecosystem.

Constraint. Constraints define domain-specific rules that govern the behavior of the DT to ensure compliance with legal, ethical, and operational standards. In the context of ITS, these constraints include traffic signal timings that must remain within operational limits, maximum vehicle density thresholds for road segments, and safety margins for pedestrian crossings. Additionally, environmental factors, such as weather conditions, impose constraints on the reliability of predictive models. By embedding these constraints into its architecture, the DT operates within predefined boundaries, ensuring safe, reliable, and regulation-compliant functionality.

Validation. The DT is validated using a multifaceted approach to ensure accuracy and reliability. Real-time sensor data collected from IoT-enabled devices at intersections are compared against DT predictions to assess model performance. Simulations are conducted to test intervention strategies, such as rerouting, under varying traffic conditions. In addition, stakeholder feedback on system recommendations provides qualitative insights, ensuring that the DT meets practical requirements and operational expectations.

Actions. Actions represent the operational behaviors assigned to domain concepts within the DT. These include dynamic signal timing adjustments, traffic rerouting based on real-time conditions, and generating predictive alerts for traffic operators. By translating the DT's insights into actionable interventions, these behaviors create a feedback loop that continuously refines the system's performance.

State Model. The State Model defines the dynamic conditions of key entities within the ITS to capture their operational status and transitions. Vehicles can be in states such as idle, moving, queuing, or rerouted. Intersections are classified as free flow, congested, or under intervention. Road segments are categorized based on traffic levels, ranging from clear to heavy congestion. The overall system state reflects conditions like normal operations, incident response, or weather-affected scenarios. These states enable real-time monitoring and adaptive decision-making within the DT framework.

Domain Concept Model (DCM). DCM defines the ITS as a set of interconnected components that capture attributes,
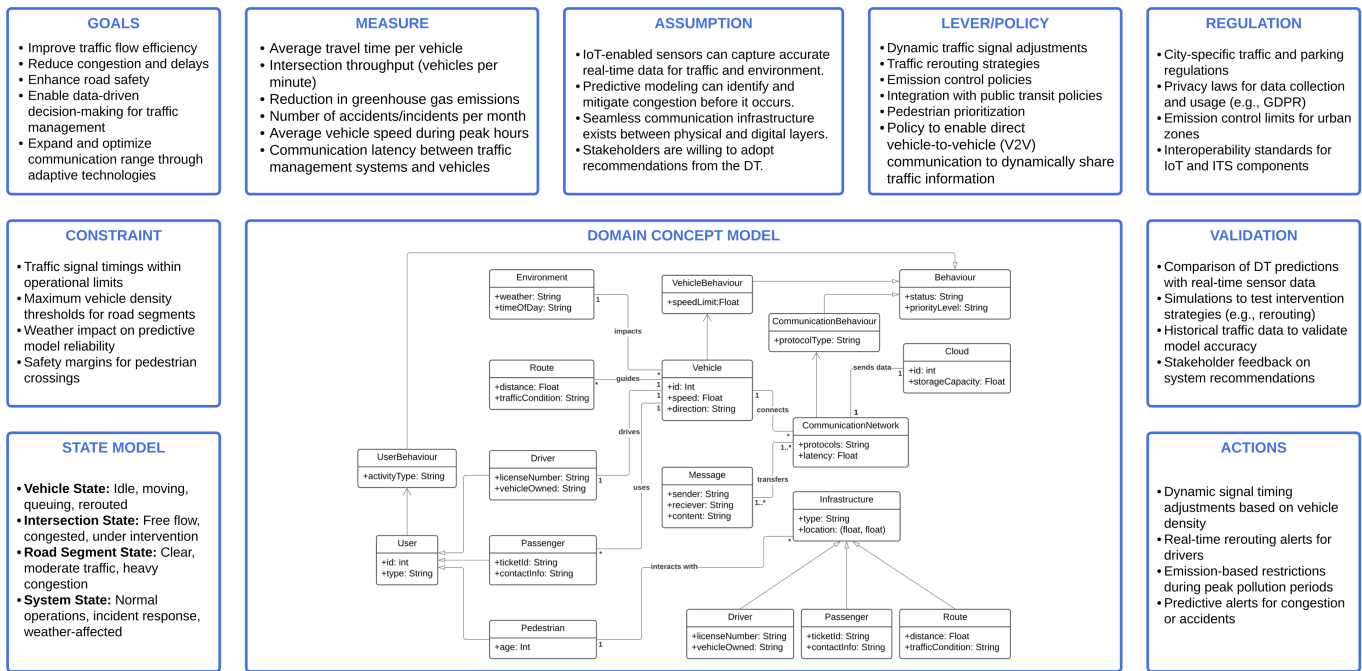
Fig. 1. Sociotechnical DT Specification Meta Language Concepts for ITS.

behaviors, and relationships at a higher level of abstraction. It provides a foundational framework for modeling key entities such as vehicles, users, communication networks, infrastructure, routes, and environmental conditions, along with their interactions. Each component in the DCM can be further elaborated through specialized domain models to address specific aspects of ITS operations in greater detail. We used the principles of *Domain-Driven Design (DDD)* to identify key domains in the ITS. This was achieved by identifying *bounded contexts* within the ITS domain, as proposed by Macías et al. [24] and Evans [25]. These bounded contexts provide logical partitions that encapsulate specific functionalities, ensuring modularity and consistency across the system.

Building on this foundation, the model organizes the ITS into several interconnected components, each corresponding to a bounded context as shown in Figure 1. For instance, the *Users Context* categorizes users as drivers, passengers, and pedestrians, each with distinct attributes and behaviors. Similarly, the *Vehicles Context* models properties such as speed, direction, and unique identifiers, as well as behaviors like adherence to speed limits, to capture real-world dynamics. The *Communication Context* defines networks and protocols to enable seamless data exchange between vehicles, infrastructure, and centralized systems, supporting real-time information sharing and decision-making. Environmental factors, such as weather and time of day, influence traffic conditions and vehicle performance, while infrastructure components define fixed resources like traffic signals and IoT-enabled sensors. Messages exchanged through communication protocols support coordination, while routes and traffic conditions guide vehicle navigation and behavior. The DCM also models interactions and constraints, including traffic signal timing limits, vehicle density thresholds, and pedestrian safety margins. These constraints are embedded within the architecture to ensure compliance with regulatory and operational requirements. Behavioral elements, such as adaptive signal timing, dynamic rerouting, and message exchanges, enable predictive analytics and scenario testing within the DT. By abstracting ITS components at a high level, DCM allows modular expansion, enabling detailed exploration of specific domains through separate models. Figure 1 illustrates the structure, showing how core entities and relationships form the basis for an extensible and scalable ITS framework. It is important to note that this DCM is not exhaustive but rather serves as an illustrative example of how the ITS domain can be modeled. The framework is flexible and can be extended or customized to incorporate additional components and interactions as needed to address the specific requirements of diverse ITS applications.

B. Proposed Architecture of the DT

Building on the DCM, we propose an architecture for $\mathcal{D}ig\mathcal{IT}$ platform that transforms the high-level abstractions defined in the DCM into operational components for data collection, predictive analysis, simulation and visualization. Although DCM provides a broad framework for modeling ITS, this architecture focuses on select components essential for real-time traffic monitoring, prediction, and scenario-based simulations. Using elements from the DCM, such as vehicles, communication networks, sensors, and user behaviors, the proposed architecture ensures a seamless integration between conceptual modeling and practical implementation. The following sections detail each functional component, demonstrating how the principles outlined in the DCM are operationalized within the architecture of $\mathcal{D}ig\mathcal{IT}$ platform.

Users and Vehicles. As shown in Figure 2, the architecture models users, including drivers, passengers, and pedestrians, along with vehicles characterized by attributes such as speed, direction, and ID. Users access dashboards that provide real-time traffic updates and predictions, extending the behavioral

modeling of users and vehicles defined in the DCM to ensure synchronization between physical and digital systems.

Sensors. As shown in Figure 2, IoT-enabled sensors deployed at intersections and road segments capture real-time data, including vehicle speeds, traffic density, and congestion levels. Environmental sensors complement this by monitoring external factors such as weather conditions, aligning with the *Environment* component in the DCM (Figure 1). These sensors operate continuously, reflecting the dynamic nature of the transportation network and providing input for predictive modeling and simulations. By integrating data from physical infrastructure, the sensors ensure that the DT maintains synchronization with real-world conditions, enabling timely analysis and decision-making.
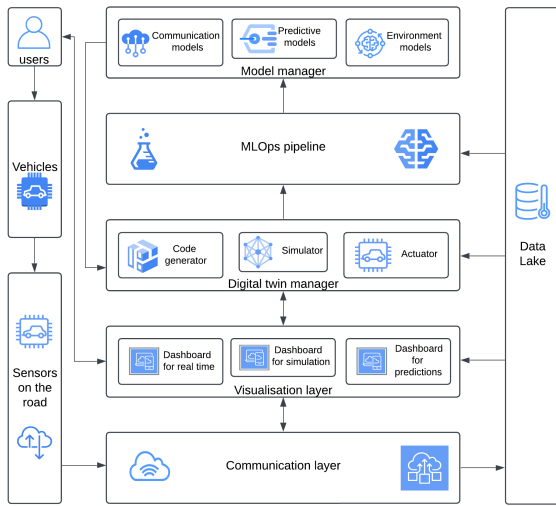

Fig. 2. Architecture of $\mathcal{D}igIT$

Communication Layer. The communication layer, as shown in Figure 2, facilitates seamless and reliable data exchange between the physical and digital systems. It leverages a heterogeneous network comprising 4G/5G, LoRA, and WiFi communication mechanisms to support high-frequency real-time data streams and periodic batch transfers. Modeled in the DCM (Figure 1) through the *CommunicationNetwork* and *CommunicationBehaviour* components, this layer ensures bidirectional communication, enabling the DT to relay recommendations—such as optimized signal timings or rerouting strategies—back to the physical infrastructure. Attributes such as protocol type and latency, as captured in the DCM, align with the low-latency protocols implemented here to handle time-critical updates, including responses to accidents or congestion. For example, vehicle counts and speed data collected at multiple intersections are transmitted to the data lake while maintaining synchronization across all input sources.

Data Lake. After data is transmitted through the communication layer, the data lake stores the communicated information, supporting both real-time analysis and long-term storage. It processes structured and unstructured data, performing preprocessing tasks like cleaning and aggregation to optimize it for simulations and machine learning models. As shown in Figure 2, this layer integrates inputs modeled in the DCM, including vehicle behavior, environmental conditions, and communication protocols.

Visualization. As shown in Figure 2, the visualization layer provides three types of dashboards—*real-time, simulation,* and *prediction*. The real-time dashboard monitors live metrics like vehicle counts and congestion levels. The simulation dashboard models scenarios such as accidents and peak-hour traffic, while the prediction dashboard forecasts future traffic patterns using historical and real-time data. These dashboards leverage the behavioral and environmental models defined in the DCM to ensure synchronization between real-world data and simulated outputs.

Digital Twin Manager. As shown in Figure 2, the DT Manager creates a virtual representation of the physical transportation network. It enables the simulation, prediction, and optimization of traffic conditions in real-time, while ensuring continuous feedback between the virtual and physical systems. The DT Manager incorporates three key components: the *Simulator*, the *Actuator*, and the *Code Generator*. These components collaboratively ensure the platform's adaptability, scalability, and responsiveness to evolving traffic conditions.

The *Simulator* replicates real-world traffic conditions by using real-time and historical data from IoT-enabled sensors, as described in the Figure 2. It models traffic dynamics such as vehicle movements, congestion patterns, and disruptions (e.g., accidents or roadblocks). Scenarios are evaluated using predictive models, such as LSTM and BiLSTM, to assess the impact of potential interventions like signal timing adjustments, lane closures, or rerouting strategies. For instance, when a sensor detects congestion at an intersection, the Simulator evaluates alternative traffic flow patterns and identifies the most effective mitigation strategy. The simulations are run iteratively, incorporating temporal and spatial dependencies, ensuring the virtual environment remains in sync with the physical system.

The *Actuator* implements decisions derived from the Simulator into the physical transportation network. By translating simulation outcomes into real-world actions, it controls physical infrastructure such as traffic lights, dynamic message signs, and variable speed limits. For example, if the Simulator predicts bottlenecks at a specific intersection, the Actuator adjusts traffic signal timings or reroutes vehicles to alleviate congestion. The Actuator is designed to handle low-latency operations, ensuring rapid deployment of interventions during critical scenarios, such as accidents or emergency evacuations. The bidirectional communication with physical systems ensures that the Actuator can validate its actions using updated feedback from sensors, maintaining a robust closed-loop system.

The *Code Generator* serves as the intermediary between the high-level abstract models defined in the DCM and the operational components of the Digital Twin. While ML models can directly process data for predictions without translation, abstract models in the DCM—such as communication protocols, behavioral models, and state transitions—must be translated into executable instructions for the Digital Twin to understand and act upon them. The Code Generator performs this crucial task, converting these specifications into machine-readable configurations that the Simulator and Actuator can execute. For example, the DCM may define the relationships between vehicles, sensors, and infrastructure; the Code Generator ensures these relationships are accurately mapped to system operations, enabling real-time simulation and execution. This modular approach ensures flexibility and scalability, allowing for the

seamless integration of new traffic management policies or sensor types without disrupting the existing system.

MLOps Pipeline. The MLOps pipeline automates the lifecycle of machine learning (ML) models, such as Long Short-Term Memory (LSTM) for traffic flow prediction or Convolutional Neural Networks (CNN) for anomaly detection. It handles data preprocessing, model training, deployment, and monitoring, ensuring adaptability to evolving traffic conditions. Let $X_t = \{x_1, x_2, ..., x_t\}$ represent input data collected over t time intervals. The objective is to forecast traffic states for a future horizon h as:

$$\hat{Y}_{t+h} = f(X_t; M_v)$$

where M_v denotes the current model version. Model performance is continuously evaluated using metrics such as Root Mean Squared Error (RMSE):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}.$$

To maintain model accuracy, retraining is triggered based on drift detection. Drift δ is defined as:

$$\delta = D(Y, \hat{Y}) > \epsilon,$$

where $D(\cdot)$ measures data or model drift, such as distributional changes in input features or output predictions, and ϵ represents a predefined threshold. When drift is detected, a new model version M_{v+1} is trained and deployed:

$$M_{v+1} = \text{Train}(X_t, Y_t).$$

The pipeline supports scalability by maintaining versioned repositories of models and automating retraining cycles. It reduces manual intervention while enabling responsiveness to traffic variations and anomalies. Inspired by Bhatt et al. [28], this approach emphasizes sustainable and self-adaptive workflows, ensuring robustness in dynamic environments.

Model Manager. As shown in Figure 2, the Model Manager organizes and stores models essential for the DT, including *communication*, *predictive*, and *environment* models, but is not limited to them. *Communication models* define protocols and behaviors for data exchange, ensuring seamless interaction between components. *Predictive models*, forecast traffic patterns and congestion based on historical and real-time data. *Environment models* account for external factors like weather and time of day, influencing traffic dynamics. These models are accessible to the Digital Twin Manager, with the *code generator* facilitating the translation of abstract models into executable formats, ensuring compatibility with the *simulator* and *actuator*.

V. Implementation

The implementation of the $\mathcal{D}igIT$ platform follows the architecture outlined in Section IV, operationalizing its components to enable real-time traffic monitoring, prediction, and visualization for ITS. By leveraging a modular and scalable design, the system ensures integration between data acquisition, predictive analytics, simulation, and user-facing dashboards.

Real-time traffic data was collected using IoT-enabled sensors deployed near the IIIT Hyderabad campus, capturing key attributes such as timestamps, flow rates, average speeds, and congestion levels. These sensors, corresponding to the *Sensors on the road* in the architecture (Figure 2), transmitted preprocessed data to a public API hosted on a Virtual Private Server (VPS). This API acted as the primary data pipeline, ensuring scalability and seamless integration of real-time and historical data. This data was further communicated to the *DT Manager* via the *Communication Layer* as shown in (Figure 2).

Traffic *Simulator*, a critical component of the *Digital Twin Manager* (Section IV), was implemented using the SUMO. SUMO utilized the processed sensor data to model transportation networks and simulate real-world scenarios, such as peak traffic hours, roadblocks, and intervention strategies. These simulations allowed for scenario-based testing of adaptive traffic management measures, including dynamic signal timing and rerouting strategies, in alignment with the behavioral models defined in the DCM (Section IV).

Predictive analytics was powered by ML models, which were handled by the *Model Manager*, as described in Section IV. Specifically, Long Short-Term Memory (LSTM) and Bidirectional LSTM (BiLSTM) networks, implemented in Python using TensorFlow, were used. These models were designed to analyze temporal patterns in traffic data and provide short-term forecasts, such as predicting congestion levels at specific intersections. The models were integrated with an *MLOps pipeline* to automate processes such as training, validation, and deployment. This pipeline continuously monitored model performance using metrics like Root Mean Squared Error (RMSE). Retraining was dynamically triggered when performance degradation exceeded predefined thresholds, ensuring sustained accuracy under evolving traffic conditions, as described in Section IV.

The *visualization layer* was implemented using a React-based frontend and a Node.js backend, connected to the API for real-time updates. The dashboard provided an intuitive interface for monitoring key metrics such as vehicle counts, average speeds, and traffic intensity. It also visualized simulation results and predictions through interactive graphs and heatmaps, reflecting the actionable insights derived from the DT's predictive and simulation engines. This implementation extended the DCM's state and behavioral models into user-facing applications, ensuring operational transparency and stakeholder engagement. The system leveraged the cloud infrastructure for scalable storage and computational resources, supporting both real-time analytics and batch processing. This infrastructure enabled seamless integration across all layers of the architecture, ensuring robust and efficient operation. Overall, the implementation demonstrated the practical realization of the proposed architecture, maintaining synchronization between physical and digital systems while addressing the complex requirements of ITS.

VI. Evaluation

The evaluation of the architecture of $\mathcal{D}igIT$ platform is guided by three key aspects:

- *Accuracy*: The predictive performance of the analytics component, measured against observed traffic flow data.
- *Fidelity*: The ability of the simulation environment to replicate real-world traffic patterns and responses to interventions.
- *Efficiency*: The computational performance of the system, including the time required to execute predictions and run simulations.

A. Data Collection and Setup

The data used for evaluating $\mathcal{D}igIT$ was collected from IoT-enabled sensors deployed at intersections near the IIIT Hyderabad campus, as described in Section III. These sensors capture traffic parameters, including vehicle counts, speeds, and congestion levels, at fixed intervals of *5 minutes*. This frequency ensures that the system captures real-time variations in traffic conditions, providing a foundation for both predictive analytics and simulations. The collected data is transmitted via the *Communication Layer* to the *Data Lake*, where it is preprocessed to handle missing values, normalize readings, and structure data for modeling. The dataset spans several days, encompassing both peak and non-peak traffic conditions, making it suitable for evaluating system performance across varying levels of congestion. Using this detailed data set, the DT ensures accurate modeling of traffic dynamics, allowing more realistic simulations and precise predictions. To develop and validate predictive models, the data set was divided into a *training set (75%)*, *validation set (15%)*, and a *testing set (10%)*. The data was structured into sequences of *15 input timesteps* to predict the *16th timestep*, reflecting the temporal dependencies required for short-term forecasting. For the current implementation, the model takes traffic flow data as input and outputs predicted flow values for future time intervals.

B. Predictive Model Validation: Accuracy

The predictive analytics component of the architecture of $\mathcal{D}igIT$ platform employs deep learning models, specifically Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (BiLSTM), implemented within the *MLOps Pipeline*, as shown in Figure 2. These models are designed to capture temporal dependencies in traffic data, enabling accurate short-term predictions of traffic flow and congestion patterns. Both models were trained using preprocessed data collected from IoT-enabled sensors, as described in Section III. The data was structured into sequences of *15 input timesteps* to predict *16th timesteps*, reflecting the temporal dependencies required for short-term forecasting. Traffic flow rates, measured in vehicles per 5 minutes, were used as the target variable to ensure precise predictions suitable for real-time decision-making and intervention strategies. For example, the models can predict increased congestion at an intersection, triggering appropriate interventions such as rerouting traffic or adjusting signal timings. The performance of the LSTM and BiLSTM models was evaluated using standard metrics, including Mean Absolute Error (MAE) (vehicles), Root Mean Squared Error (RMSE) (vehicles), and Mean Absolute Percentage Error (MAPE) (percentage). The results are presented in Table I.

TABLE I
PERFORMANCE METRICS FOR PREDICTIVE MODELS

Model	RMSE (vehicles)	MAE (vehicles)	MAPE (%)
LSTM	25.522	18.394	13.3
BiLSTM	24.451	17.255	19.1

The results demonstrate that both models effectively capture traffic dynamics. The BiLSTM model achieved slightly lower RMSE and MAE values, indicating higher predictive accuracy. However, the LSTM model exhibited a lower MAPE, suggesting it may be more robust for percentage-based error evaluation.

These findings highlight the complementary strengths of the two models, depending on specific performance criteria. Predictions generated by the models were visualized through the *Visualization Layer* using the *Dashboard for predictions*, as shown in Figure 3, providing insights into both immediate and future traffic patterns. These insights allow stakeholders to assess the effectiveness of interventions in real-time, ensuring timely and informed traffic management decisions.
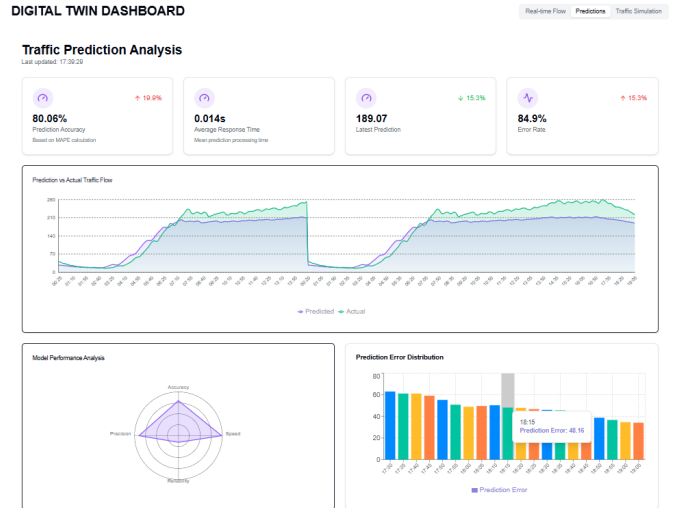

Fig. 3. Traffic Prediction Dashboard displaying accuracy, prediction errors, and performance metrics.

C. Simulation Fidelity

The simulation fidelity of the architecture of $\mathcal{D}igIT$ platform was evaluated by analyzing its ability to replicate observed traffic patterns and predict traffic flow dynamics accurately. This evaluation combined machine learning predictions with simulation outputs to validate the system's ability to model real-world traffic behavior effectively. Traffic predictions were generated using Long Short-Term Memory (LSTM) and Bidirectional LSTM (BiLSTM) models, implemented within the *MLOps Pipeline*. These predictions were compared against actual traffic flow data collected at 5-minute intervals from IoT-enabled sensors, as shown in Figure 4. The results demonstrated a strong correlation between predicted and observed values, capturing key traffic features such as congestion buildup, clearance, and peak-hour variations. To ensure consistency, the modeling assumptions between SUMO simulations and the DT were carefully aligned, including vehicle flow rates, signal timing protocols, and road network configurations. For example, vehicle flow rates from IoT sensors were matched with the input parameters in SUMO to reflect observed traffic volumes. This alignment ensures that the virtual simulations reflect real-world dynamics as closely as possible. Some spikes were observed during periods of high congestion, as shown in Figure 4, due to insufficient data for such scenarios; however, the model is expected to improve with additional data collection. Simulations were executed using the SUMO platform, integrated within the *Simulation and Digital Twin Manager*, to model virtual scenarios reflecting real-world

dynamics. These simulation outputs were visualized in the *Visualization Layer* via an interactive dashboard, providing insights into traffic behavior and intervention strategies, as shown in Figure 5.
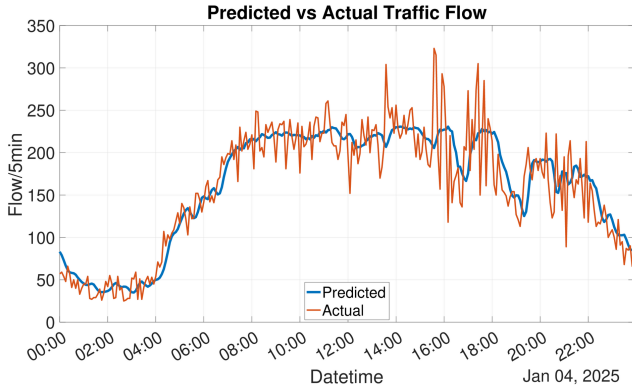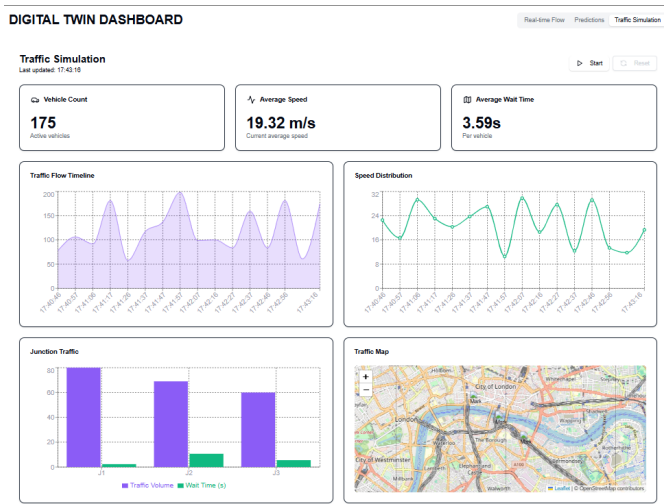

Fig. 4. Predicted vs Actual Traffic Flow.

Fig. 5. Simulation Dashboard displaying traffic flow timelines, speed distributions, and junction traffic visualizations.

D. Computational Efficiency

The computational performance of $\mathcal{D}igIT$ was evaluated to validate its suitability for real-time applications. The assessment measured execution times for both predictive modeling and simulation tasks, ensuring responsiveness under time-sensitive conditions. The LSTM and BiLSTM models, implemented within the *MLOps Pipeline*, required an average of *7 milliseconds* to process 15 input timesteps and generate predictions for the next 16 timesteps. Simulations covering the same interval were executed within *15 seconds* using SUMO. These results demonstrate that the system operates within time constraints appropriate for real-time traffic management, enabling timely responses to evolving traffic conditions. The integration of the *MLOps Pipeline* further enhances computational efficiency by automating the retraining and deployment of predictive models. This process ensures that the system adapts dynamically to changes in traffic behavior without manual intervention, maintaining both accuracy and scalability as data volumes increase.

VII. Discussion and Conclusion

This paper presented the architecture for the $\mathcal{D}igIT$ platform. The platform integrates predictive modeling, simulations, and automated workflows to address real-time traffic management challenges. The *DCM* models components such as vehicles, sensors, and communication networks, ensuring seamless data flow between physical and digital systems.

Evaluation results demonstrated that the platform effectively captured traffic patterns and provided accurate short-term forecasts. Simulations were used to validate the computational efficiency of the platform, showing an average prediction time of 7 ms. The modular design supports scalability, enabling integration of advanced modeling techniques, such as reinforcement learning and hybrid simulations, to enhance adaptability and performance. While the current implementation focuses on traffic forecasting and data-driven modeling, the architecture provides a foundation for expanding capabilities to address broader challenges, including multi-modal transportation systems and communication network modeling. The results validate the feasibility of Digital Twins for traffic management and highlight their potential for improving efficiency and decision-making in dynamic transportation environments.

VIII. Future Work

Future efforts will focus on extending the architecture for $\mathcal{D}igIT$ platform to support multi-modal transportation systems, including public transit and pedestrian flows. Enhancements in modeling communication networks will enable evaluations of key metrics such as latency, throughput, and range, improving performance in connected and autonomous vehicle scenarios. Additional focus will be on integrating adaptive modeling techniques and dynamic calibration methods to enhance scalability during peak traffic and disruptions. Testing across diverse traffic scenarios, including urban roads, highways, and mixed-mode networks, will validate generalizability. Finally, advancements in real-time analytics leveraging IoT and 5G technologies will be explored to strengthen data acquisition, communication reliability, and responsiveness, ensuring the platform can handle dynamic traffic environments effectively.

IX. Acknowledgments

References

[1] A. Perallos, U. Hernandez-Jayo, E. Onieva, and I. J. G. Zuazola, *Intelligent transport systems: technologies and applications*. John Wiley & Sons, 2015.

[2] J. Andersen and S. Sutcliffe, "Intelligent transport systems (its)-an overview," *IFAC Proceedings Volumes*, vol. 33, no. 18, pp. 99–106, 2000.

[3] "Early estimates of motor vehicle traffic fatalities and fatality rate by sub-categories through june 2024." https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/813661, 2024.

[4] "Road accidents in india 2022." https://morth.nic.in/sites/default/files/RA_2022_30_Oct.pdf, 2022. Accessed: 2022-10-30.

[5] M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," *Transdisciplinary perspectives on complex systems: New findings and approaches*, pp. 85–113, 2017.

[6] M. Liu, S. Fang, H. Dong, and C. Xu, "Review of digital twin about concepts, technologies, and industrial applications," *Journal of manufacturing systems*, vol. 58, pp. 346–361, 2021.

[7] B. R. Barricelli, E. Casiraghi, and D. Fogli, "A survey on digital twin: Definitions, characteristics, applications, and design implications," *IEEE access*, vol. 7, pp. 167653–167671, 2019.

[8] H. Feng, H. Lv, and Z. Lv, "Resilience towarded digital twins to improve the adaptability of transportation systems," *Transportation Research Part A: Policy and Practice*, vol. 173, p. 103686, 2023.

[9] M. Jafari, A. Kavousi-Fard, T. Chen, and M. Karimi, "A review on digital twin technology in smart grid, transportation system and smart city: Challenges and future," *IEEE Access*, vol. 11, pp. 17471–17484, 2023.

[10] Z. Wang, R. Gupta, K. Han, H. Wang, A. Ganlath, N. Ammar, and P. Tiwari, "Mobility digital twin: Concept, architecture, case study, and future challenges," *IEEE Internet of Things Journal*, vol. 9, no. 18, pp. 17452–17467, 2022.

[11] S. Hochreiter, "Long short-term memory," *Neural Computation MIT-Press*, 1997.

[12] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The performance of lstm and bilstm in forecasting time series," in *2019 IEEE International Conference on Big Data (Big Data)*, pp. 3285–3292, 2019.

[13] G. Symeonidis, E. Nerantzis, A. Kazakis, and G. A. Papakostas, "Mlops - definitions, tools and challenges," in *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, 2022.

[14] L. Kanigolla, G. Pal, K. Vaidhyanathan, D. Gangadharan, and A. Vattem, "Architecting digital twin for smart city systems: A case study,"

[15] E. Glaessgen and D. Stargel, "The digital twin paradigm for future nasa and us air force vehicles," in *53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA*, p. 1818, 2012.

[16] N. Mohammadi and J. E. Taylor, "Smart city digital twins," in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1–5, 2017.

[17] S. Barat, A. Paranjape, R. Parchure, S. Darak, and V. Kulkarni, "Agent based simulatable city digital twin to explore dynamics of covid-19 pandemic," in *2022 Winter Simulation Conference (WSC)*, pp. 557–568, IEEE, 2022.

[18] B. S. Barn, T. Clark, S. Barat, and V. Kulkarni, "Towards the essence of specifying sociotechnical digital twins," in *Proceedings of the 16th Innovations in Software Engineering Conference*, ISEC '23, (New York, NY, USA), Association for Computing Machinery, 2023.

[19] A. Redelinghuys, A. H. Basson, and K. Kruger, "A six-layer architecture for the digital twin: a manufacturing case study implementation," *Journal of Intelligent Manufacturing*, vol. 31, no. 6, pp. 1383–1402, 2020.

[20] V. Kulkarni, S. Barat, and T. Clark, "Towards adaptive enterprises using digital twins," in *2019 Winter Simulation Conference (WSC)*, pp. 60–74, 2019.

[21] L. Bao, Q. Wang, and Y. Jiang, "Review of digital twin for intelligent transportation system," pp. 309–315, 10 2021.

[22] K. Kušić, R. Schumann, and E. Ivanjko, "A digital twin in transportation: Real-time synergy of traffic data streams and simulation for virtualizing motorway dynamics," *Advanced Engineering Informatics*, vol. 55, p. 101858, 2023.

[23] J. Lee, B. Bagheri, and H.-A. Kao, "A cyber-physical systems architecture for industry 4.0-based manufacturing systems," *Manufacturing letters*, vol. 3, pp. 18–23, 2015.

[24] A. Macías, E. Navarro, C. E. Cuesta, and U. Zdun, "Architecting digital twins using a domain-driven design-based approach," in *2023 IEEE 20th International Conference on Software Architecture (ICSA)*, pp. 153–163, IEEE, 2023.

[25] E. Evans, *Domain-driven design: tackling complexity in the heart of software*. Addison-Wesley Professional, 2004.

[26] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*, IEEE, 2018.

[27] Y. Dai and Y. Zhang, "Adaptive digital twin for vehicular edge computing and networks," *Journal of Communications and Information Networks*, vol. 7, no. 1, pp. 48–59, 2022.

[28] H. Bhatt, S. Arun, A. Kakran, and K. Vaidhyanathan, "Towards architecting sustainable mlops: A self-adaptation approach," *arXiv preprint arXiv:2404.04572*, 2024.