JOURNAL OF
Emerging Paradigms
in Computing Systems
JEPCS

Journal of Emerging Paradigms in Computing Systems (JEPCS) | Volume 02, Issue No. 01, 2026

ARTICLE TYPE: Review Article

Article ID: JEPCS-2026-0201-0028

IoT and Digital Twins for Enterprise Digital Transformation: A Structured Review of Architectures, Enabling Technologies, Applications, and Research Challenges

Saira Shairi[1,*], Saqib Dar[2]

[1] Department of Software Engineering, The University of Lahore, Lahore, Pakistan
[2] Independent Researcher, Riyadh, Saudi Arabia

*Corresponding author: saira.shairi@se.uol.edu.pk
Contributing authors: saira.shairi@se.uol.edu.pk, saqi2926@gmail.com

Received: October 26, 2025 **Accepted:** December 27, 2025

Abstract- Enterprises that operate physical assets increasingly use IoT devices to collect operational data and Digital Twins (DTs) to represent the condition of assets and processes in digital form. IoT provides measurements from the physical environment and, in some applications, also carries commands back to equipment. A DT uses these data to maintain and update a representation of the corresponding physical entity. Previous reviews have usually examined DT architecture, IoT, edge/cloud computing, artificial intelligence, cybersecurity, or individual application domains as separate topics. In this paper, we review 45 peer-reviewed papers between 2017 and 2025 and examine how these technologies are combined in enterprise DT systems. Each paper was examined according to its architectural role, enabling technologies, application domain, decision function, and reported technical limitations. The comparison resulted in a seven-layer architecture covering the physical and IoT layer, communication, edge/fog processing, cloud and enterprise data, DT models and services, intelligence and decision support, and enterprise applications and governance. We also identified five forms of DT integration. They range from systems used mainly for monitoring to implementations in which DT information is connected to operational decisions or control. Across the reviewed papers, recurring problems include interoperability, synchronization between physical and digital states, model validity, cybersecurity, lifecycle management, computational requirements, and integration with existing enterprise systems. The review indicates that the usefulness of an enterprise DT depends on how well physical observations, the digital representation, decision functions, and operational actions are connected within the same system.

Keywords- Digital transformation; Digital twin; Internet of Things; Edge and cloud computing; Enterprise architecture

1. Introduction

Replacing a manual activity with a digital interface does not by itself constitute digital transformation. Digital transformation involves changes in how an organization collects and uses information, coordinates its processes, makes decisions, and creates value. These changes may involve technology, but they also affect operational procedures and the way different parts of an organization interact. Vial described digital transformation as a process in which digital technologies create disruptions that trigger strategic responses and changes in value creation paths [1]. Verhoef et al. similarly separated digitization, digitalization, and digital transformation, and emphasized that transformation has implications beyond the information technology function [2]. Nadkarni and Pruegl showed that the concept spans technological and organizational research streams and therefore needs a clear link between technical capability and organizational change [3]. These

distinctions matter in cyber-physical enterprises because collecting more data does not, by itself, change how an organization operates. One component of the connection is provided by the Internet of Things (IoT). Physical process states can be recorded by sensors, embedded controllers, mobile devices, machines, and gateways, which can then share data with nearby or distant computing systems. IoT was positioned as a technology foundation for linked industrial processes alongside cloud computing, cyber-physical systems, analytics, and automation in Industry 4.0 research [4]. However, an IoT implementation primarily addresses issues related to actuation, observation, and communication. A stream of measurements does not automatically provide a model of the physical entity, explain how its state is evolving, or support a controlled evaluation of future actions. DT research addresses this second requirement. Early manufacturing work distinguished a digital model, a digital shadow, and a DT according to the direction and degree of automated data exchange between physical and digital entities [5]. Later papers described DTs as combinations of physical entities, virtual models, data, services, and connections that can support monitoring, simulation, prediction, and control [6, 7]. In practice, IoT and DTs perform different but connected functions. IoT devices report the state of the physical system through measurements collected from sensors and controllers. A DT relates these measurements to a model of the corresponding asset or process. In applications that support decision making or control, information may also move in the opposite direction. A recommendation may be returned to an operator, while applications with actuation capability can send a command to a controller or physical device. Previous DT surveys have approached the field from different directions. Previous surveys examine different parts of the DT research space. Fuller et al. [8] discuss enabling technologies and open research issues, while Jones et al. [9] focus on DT definitions and classification. Industrial applications are reviewed by Liu et al. [10], and Qi et al. [11] examine technologies and tools used in DT implementation. Qian et al. [12] examine the architectural organization of DT systems. Sharma et al. [13] consider the relationship between DT theory and practice, while Liu et al. [14] organize DT systems around physical entities, virtual models, twin data, and applications. More focused reviews address particular technical or application concerns. Hakiri et al. [15] examine DTs in future networks and IoT environments. Cybersecurity is considered from general DT and industrial perspectives in [16–18], and production-related implementation challenges are discussed in [19]. Standards and interoperability are examined in [20, 21]. Other reviews focus on DT Networks [22], smart-city applications [23], IIoT and Industry 5.0 [24], Industry 4.0 reference architectures [25], and DT-assisted edge offloading [26]. The surveys summarized above are organized around different technologies, architectural concerns, or application domains. Across this selected set, the full path from physical observation to enterprise use is not consistently treated as a single organizing frame. In an operational system, data may pass through sensing devices, communication networks, edge or cloud resources, DT models, analytical functions, and enterprise software before contributing to a decision. This review examines how information remains associated with the correct physical entity and how state is maintained as data move through these parts of the system. The physical-to-enterprise path is therefore used as the organizing perspective, from observation and DT processing to decision support and, where applicable, operational action. In this review, *enterprise* refers to an organization that operates physical or operational systems and connects them with digital technologies. The represented environment may include assets, facilities, infrastructure, products, logistics networks, or human-centered services. The application scope covers manufacturing, healthcare, smart cities and infrastructure, buildings and energy systems, agriculture, and supply chains. Purely virtual consumer applications are excluded unless they contribute an architectural concept that can be transferred to a cyber-physical enterprise setting. The review addresses six research questions:

- RQ1: What roles do IoT and DTs perform in enterprise digital transformation?
- RQ2: How can physical assets, IoT data, computing resources, DT models, decision functions, and enterprise applications be organized within a common architecture?
- RQ3: Which enabling technologies recur in IoT–DT systems, and where do they operate within the architecture?
- RQ4: How do application domains differ in the entity being represented, the decision being supported, and the operational requirements placed on the DT?
- RQ5: Which technical and governance barriers recur in DT deployment, scaling, and reuse?
- RQ6: What problems remain when separate DT implementations are connected into interoperable and governed enterprise systems?

The analysis links IoT, DTs, edge/cloud computing, AI, and enterprise systems along the path from physical observation to operational use. From this analysis, the paper defines a seven-layer architecture that separates the responsibilities of physical devices, communication, computing resources, DT models, decision functions, and enterprise applications. The 45 reviewed papers are also grouped into five integration patterns according to how DT information is used for monitoring, diagnosis, prediction, decision support, and closed-loop action. Reported deployment barriers are organized into technical and governance categories and are used to identify research directions involving interoperability, uncertainty, cybersecurity, lifecycle management, sustainability, and enterprise-level evaluation.

The rest of the paper is organized as follow: Section 2 defines the main concepts and positions the paper relative to earlier DT surveys. Section 3 describes the review design, paper selection, and coding procedure. Section 4 examines the technical foundations while Section 5 presents the seven-layer architecture. Section 6 discusses the enabling technologies and integration patterns. Section 7 compares the application domains, and Section 8 examines the cross-domain research

challenges. Section 9 presents the research agenda. Section 10 discusses the limitations of the review, and Section 11 concludes the paper.

2. Background and Positioning of the Review

2.1. Digital transformation as a technical and organizational process

A useful starting point is to separate three terms that are often used as if they were interchangeable. *Digitization* converts information from analog to digital form. *Digitalization* uses digital technologies to improve an existing activity or process. *Digital transformation* changes how the organization coordinates resources, makes decisions, provides services, or creates value [1, 2]. This distinction prevents a common classification problem. Installing sensors on a machine is an IoT project. Connecting the sensor feed to a maintenance model can digitalize maintenance. A broader transformation occurs when the maintenance process, parts planning, production scheduling, accountability, and performance measurement are reorganized around the new information flow. For cyber-physical enterprises, transformation therefore has at least two coupled dimensions. The first is technical: physical state must be captured, transported, interpreted, modeled, and made available to applications. The second is operational: the resulting information must change a decision, workflow, control action, or service. IoT and DTs connect the physical and digital parts of an enterprise system. IoT devices collect measurements from physical assets and processes. A DT uses these measurements to update a digital representation that can support simulation, prediction, or decision support. In applications that include control, information can also be returned from the digital system to an operator, controller, or actuator.

2.2. From connected assets to DTs

IoT systems use different sensing, communication, and computing mechanisms. Their devices may differ in sampling rate, processing capability, energy availability, communication protocol, and ownership. Research on organizational IoT adoption shows that deployment issues are not limited to sensing and connectivity. Compatibility, security, skills, cost, governance, and integration with existing processes also affect adoption [27]. Manufacturing research similarly indicates that Industry 4.0 applications commonly depend on combinations of technologies rather than on a single technical component [28]. The distinction between a connected asset and a DT depends partly on how data move between the physical and digital entities. Kritzinger et al. [5] separated digital representations into three forms. A digital model has no automatic exchange of data with its physical counterpart. Data is automatically moved from the real object to its digital version in a digital shadow. By facilitating automatic trade in both ways, a DT expands the physical-to-digital link. The information transfer between the actual object and its digital representation is the primary difference. Visualization can be supported by a live dashboard or a three-dimensional model, but a DT needs a maintained representation that is updated from the physical system and when necessary can enable information flow back to it. This representation is described in a variety of ways in the literature. Shop-floor DTs are framed around resources, goods, processes, and services by Tao and Zhang [7]. Qi and Tao [6] concentrate on the relationship between DTs and big data in smart manufacturing. Qian et al. [12] describe the DT through its architecture, data connection, and applications. Liu et al. [14] use a structure based on physical entities, virtual models, twin data, and applications. These formulations differ in emphasis, but all rely on an identifiable physical device and a digital representation that is updated through data exchange. Although these formulations use different structures, each depends on an identifiable physical device and a mechanism for updating its digital representation. Synchronization frequency, model detail, and actuation capability then vary according to the intended application.

2.3. Role of Edge, Cloud, AI, and the Digital Thread

Data used by a DT may be generated continuously and at different sampling rates. Transmitting every measurement to a remote cloud can add communication traffic and delay. Edge and fog resources allow selected processing to take place closer to the source. Filtering, aggregation, local analysis, and time-sensitive decision functions can therefore be performed before data are transferred to higher-level services. Research on fog-based IoT systems identifies location awareness, distributed processing, mobility support, and reduced bandwidth use among the reasons for placing computation near end devices [29]. DT-assisted edge systems similarly use nearby computing resources for time-sensitive twin functions and computation offloading [26]. Cloud platforms support functions that require greater storage, computation, or access to data collected over longer periods. Cloud platforms support storage and computation that may be difficult to provide close to the physical system. They can retain long-term operational data, support model training and model repositories, perform analysis across multiple assets, and connect DT services with enterprise applications. Placement depends on the requirements of the individual function, including response time, communication cost, privacy, reliability, and lifecycle management. Processing with tight timing constraints may be kept near the physical process. Larger datasets or more computationally demanding tasks can be carried out on cloud or business systems. When the DT must estimate, analyze, or forecast data that cannot be obtained directly from measurements, AI becomes important. State estimation, anomaly detection, forecasting, optimization, perception, and decision assistance are some potential applications. Applications of AI in Industry 4.0 papers include forecasting, process control, quality evaluation, and predictive maintenance [30]. The same review literature identifies data quality, model interpretability, system integration, and deployment as continuing

implementation concerns. The digital thread links information produced at different stages of an asset or product lifecycle. A DT is generally concerned with the current or evolving state of a particular entity or process. The digital thread connects that representation with information generated during activities such as design, production, operation, and service. Cloud-based digital-thread research reports difficulties in maintaining traceability across heterogeneous engineering and enterprise data sources [31]. An operational DT can therefore be available for one process even when related engineering, maintenance, service, or compliance information remains distributed across separate systems.

2.4. Position Relative to Earlier Surveys

Table 1 compares DT reviews across the technical areas considered in this paper. The comparison includes DT concepts, IoT and networking, edge/cloud computing, security and standards, cross-domain applications, and the main emphasis of each review.

Table 1: *Comparison of the present review with DT surveys.*

Paper	DT concept	IoT / network	Edge / cloud	Security / standards	Cross-domain apps.	Primary emphasis
Fuller et al. [8]	Yes	Yes	Yes	Partial	Yes	Enabling technologies, challenges, and open research
Jones et al. [9]	Yes	Partial	Partial	Partial	Partial	Characterization of DT definitions and properties
Qi et al. [11]	Yes	Yes	Yes	Partial	Partial	Technologies and tools that enable DT implementation
Sharma et al. [13]	Yes	Yes	Yes	Partial	Yes	Theory, practice, challenges, and research questions
Liu et al. [14]	Yes	Yes	Partial	Partial	Yes	Physical entity, virtual model, twin data, and applications
Hakiri et al. [15]	Yes	Yes	Yes	Yes	Partial	Future networks and emerging IoT industry
Homaei et al. [16]	Yes	Yes	Partial	Yes	Yes	AI-assisted cybersecurity for DTs
Jeremiah et al. [17]	Yes	Yes	Yes	Yes	Yes	Applications, technologies, and security challenges
Mon et al. [24]	Yes	Yes	Yes	Yes	Partial	IIoT practices and Industry 5.0 directions
Filho and Canciglieri [25]	Yes	Yes	Yes	Partial	Partial	Industry 4.0 review and reference architecture
This review	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Enterprise integration layers, patterns, governance, and transformation outcomes**

The analytical frame used in this review follows the path from physical observation to an enterprise decision. For each paper, the analysis considers where an observation originates, where processing takes place, and how the resulting information contributes to the DT state. It also identifies the analytical or decision function that uses this state and the enterprise process in which the output is applied. When an output is connected to an operational action, decision authority and relevant governance conditions are also recorded. Using the same analytical frame across the included papers provides a consistent basis for comparing technical layers and application domains.

3. Review Methodology

3.1. Review Design and Scope

A structured literature review was used to examine how IoT, DTs, computing infrastructure, and enterprise applications are connected. Because the selected publications differ in research design, application setting, and evaluation method, the synthesis focuses on architectural characteristics rather than pooled numerical outcomes. The coding considers system roles, interfaces, application functions, and reported deployment barriers. The review covers peer-reviewed journal and conference publications available from 2017 through 2025. The starting year was selected to include the period in which explicit DT architectures and related Industry 4.0 formulations were appearing in the literature [4, 7]. Search terms were selected around the concepts examined in the review. Organizational-change terms included *digital transformation* and enterprise transformation, while DT-related searches used DT and digital shadow. Supporting technologies were represented by *Internet of Things*, IIoT, edge computing, fog computing, cloud computing, AI, standards, and cybersecurity. The search terms were also combined with the application domains covered in the review, including manufacturing, healthcare, smart cities, energy, agriculture, and supply chains. Queries were "Digital Twin AND IoT", "Digital Twin AND edge computing", "Digital Twin AND enterprise architecture", and "Digital Twin AND cybersecurity". Domain-specific searches included "Digital Twin AND healthcare" and "Digital Twin AND smart agriculture". Candidate papers were identified through publisher databases and scholarly indexing services. Their bibliographic details were checked, and each paper was examined for relevance to the scope of the review before inclusion. The resulting corpus was intentionally bounded for architectural and cross-domain analysis. Papers were screened against the research questions, after which 45 papers were retained for detailed review.

3.2. Eligibility Criteria

A paper was included when all of the following conditions were satisfied:

- the publication was a peer-reviewed journal or conference paper available by the end of 2025

- the paper addressed digital transformation, IoT/IIoT adoption, DT concepts or architectures, DT enabling technologies, DT security or standards, or a concrete DT application
- the contribution could be assigned to at least one architecture layer, integration pattern, application domain, or research challenge used in the review
- the publication provided enough bibliographic and technical detail to determine its role in the analysis.

Duplicate versions, short editorial or news items, promotional material, and papers that used the term DT without describing a physical-to-digital relationship were excluded. General digital-transformation papers were retained only when they helped define the concept or explained how a technical capability was connected to organizational change. Application papers were sampled across domains to avoid allowing manufacturing, the historically dominant DT domain, to determine the full enterprise architecture.

3.3. Analysis procedure

Each included publication was analyse along five dimensions. Architectural role records where the contribution sits in the physical-to-enterprise stack. Enabling technology records the main technologies required by the contribution, such as IoT, edge/fog, cloud, AI, data models, security mechanisms, or standards. Application function records whether the twin is used for monitoring, simulation, prediction, optimization, decision support, or control. Domain records the physical context. Barrier records the principal deployment issue reported or implied by the paper. The analysis was completed in three steps. First, common responsibilities were grouped into architecture layers. Second, application papers were compared to determine how far information travels through those layers before a decision is made. This produced the five integration patterns in Section 6. Third, recurring barriers were mapped back to the interfaces at which they occur. This last step is important because a problem such as interoperability has a different meaning at a device interface, a model interface, and an enterprise application interface. Figure 1 summarizes the review procedure used in this paper. Because the review was based on a bounded set of papers selected for architectural analysis, the figure shows the screening and coding stages without presenting exhaustive database-retrieval counts.

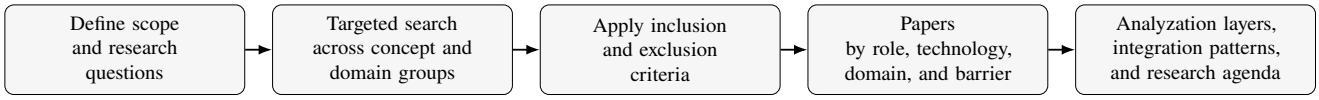


Figure 1: *Review procedure used to identify architectural roles, integration patterns, and cross-domain issues.*

The analysis is used to compare architectural roles, interfaces, and deployment problems across the selected papers. Its results are qualitative rather than statistical. Effect size, technology adoption rate, and publication frequency across the complete DT literature are therefore outside the scope of this review. Section 10 discusses this limitation in more detail.

4. Technical Foundations of IoT-DT Transformation

4.1. Observation, Identity, and State Acquisition

A DT needs a reliable association between an observation and the physical entity that the observation describes. The entity varies with the application. The physical devices can vary considerably across applications. It may be a machine or product in manufacturing, a patient or building zone in a human-centered or built environment, a vehicle, an agricultural field, a warehouse resource, or part of a larger distributed system. IoT infrastructure provides the connection to these entities through sensing and identification, with communication mechanisms carrying the resulting observations to other parts of the system. For the architectural analysis in this review, each observation is considered together with four pieces of contextual information: the identity of the physical devices, the source of the measurement, the time of observation, and information about measurement quality. These details establish which entity the measurement describes and whether it is appropriate for representing that entity at the relevant time. In enterprise environments, maintaining this association can become difficult when the same physical asset is represented by different identifiers in different systems. A machine, for example, may be identified differently by operational technology and maintenance systems. Enterprise resource planning software and engineering databases may use other identifiers for that same physical equipment. Sensors can also operate at different sampling rates or use clocks that are not perfectly synchronized. Maintaining an accurate digital state therefore requires both reliable timing information and a data model that links observations to the correct physical entity. IoT adoption papers report compatibility, security, cost, skills, and organizational readiness among the factors that affect deployment [27]. Compatibility is also relevant to DT integration. A proprietary device interface may provide enough information for a local application while making the same data difficult to reuse in a cross-vendor DT or enterprise data model. The proposed architecture therefore keeps physical connectivity separate from the higher-level representation maintained by the DT.

4.2. State Synchronization and Twin Fidelity

Synchronization describes how current the digital state is relative to the physical system. The DT may update this state directly from sensor measurements, estimate it from several observations, or combine current data with historical and

contextual information. The acceptable delay and estimation error depend on the decision for which the twin is being used. A DT may update its state directly from measurements, estimate an unobserved state from sensor data, or combine current measurements with historical and contextual information. The acceptable delay and estimation error depend on the decision for which the twin is being used. Update requirements can differ considerably between applications. A protection or control function can require frequent state updates, whereas planning applications may operate with slower data. Higher update rates also increase communication, storage, and processing requirements. Synchronization frequency should therefore be selected according to the timing and accuracy requirements of the application. Network-oriented DT papers treat communication delay, edge placement, and resource allocation as important factors for maintaining time-sensitive twin services [15, 26]. The level of model detail must be selected in the same way. A detailed finite-element model may support engineering analysis but may be too computationally expensive for continuous online execution. Reduced-order and data-driven models can lower execution cost, although their accuracy and validity depend on the operating conditions for which they were developed. An enterprise DT may therefore maintain more than one model for the same physical entity. In such cases, the purpose, operating range, provenance, and update history of each model should be recorded.

4.3. Edge, Fog, and Cloud Placement

DT functions can be distributed across device, edge, fog, and cloud resources. Edge and fog nodes can filter measurements, detect events, compress data streams, translate protocols, or maintain local state estimates. These functions are useful when communication delay or connectivity limits the use of remote services. Fog-computing research identifies low latency, geographic distribution, mobility support, and bandwidth saving as important properties for IoT workloads [29]. DT systems that operate over IoT infrastructure are subject to the same resource and communication constraints. Cloud platforms support functions with different requirements. They provide scalable storage and computation, model training, long-term history, fleet-level analysis, and interfaces to enterprise applications. Function placement should therefore reflect the requirements of the individual task. Time-sensitive or privacy-sensitive processing may remain close to the physical process, whereas computation that depends on aggregation, large models, or long-term data can be placed on cloud or enterprise platforms. A single DT can consequently use edge inference for local anomaly detection, cloud resources for model training, and an enterprise system for maintenance planning.

4.4. Intelligence and Decision Functions

The amount of intelligence required by a DT depends on its intended function. A twin that maintains and displays the current state may not require a machine-learning model. Other applications use AI for image interpretation, remaining useful life estimation, demand forecasting, anomaly detection, optimization, or policy learning. Industry 4.0 reviews report AI use across prediction, quality assessment, maintenance, and process control, together with deployment problems involving data quality, explainability, integration, and model operation [30]. The DT and the analytical model attached to it should be treated as separate components. The DT maintains information about the physical entity, its current state, and its context. An AI model can consume selected twin data and return a classification, estimate, forecast, or recommended action. This separation allows the analytical model to be validated, replaced, retrained, or disabled without rebuilding the complete DT. It also makes it possible to compare learned models with rule-based or physics-based alternatives while keeping the surrounding DT architecture largely unchanged.

4.5. Security, Trust, and Standards

A DT connects physical operations with sensing devices, networks, data stores, computational models, and decision functions. Information and commands pass through several interfaces along this path, creating multiple points at which their integrity or availability may be affected. Measurements may be modified, delayed, disclosed, or made unavailable. Commands returned to the physical system may also be altered before reaching a controller or actuator. Within the architecture considered in this review, security includes confidentiality, integrity, and availability, together with identity management and protection of the control path. Sensor measurements, model parameters, and synchronization messages influence the state maintained by the DT. If these inputs are corrupted or altered without authorization, the digital representation may no longer correspond to the physical system. Control commands require separate protection because unauthorized modification can cause the physical system to perform an action different from the one intended by the decision function. DT cybersecurity papers report threats involving communication, data, identity, models, and control functions [16–18]. The consequences are greater when DT output is used to influence a physical process because an incorrect digital state can then affect an operational action. Enterprise DTs also need to exchange information across vendor platforms, engineering tools, operational systems, and enterprise applications. This creates an interoperability problem at both the interface and information-model levels. DT standards papers describe work on manufacturing models, interfaces, information structures, and lifecycle representation [20, 21]. Standardized interfaces can reduce the amount of system-specific integration required between platforms. Differences in domain semantics, data ownership, and organizational policy still need to be handled at higher levels of the architecture. Figure 2 summarizes these relationships through an observation-model-decision-action process.

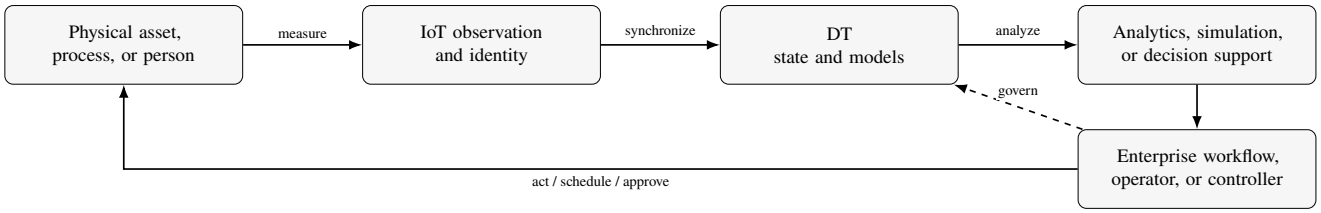


Figure 2: *IoT-DT information flow from physical observation to decision and operational action.*

5. Seven-Layer Architecture for Enterprise Digital Transformation

Layer boundaries differ across DT papers because each architecture emphasizes a different part of the system. Tao and Zhang [7] separate shop-floor resources, products, processes, and services in a manufacturing setting. Hakiri et al. [15] place greater emphasis on device, edge, and cloud functions in network-oriented DT systems. The differences among DT architectures are partly explained by the application and technical problem for which each architecture was developed. Jeremiah et al. [17], for example, include trust boundaries and control functions in their treatment of DT security. The enterprise view adopted in this review also includes lifecycle management and integration with operational applications. Figure 3 organizes these responsibilities into seven logical layers. The layers represent system functions, while their physical deployment may vary across devices, sites, and computing platforms. Functions assigned to adjacent layers may run on the same device or platform. Conversely, the functions of one logical layer may be distributed across several devices, sites, or computing platforms.

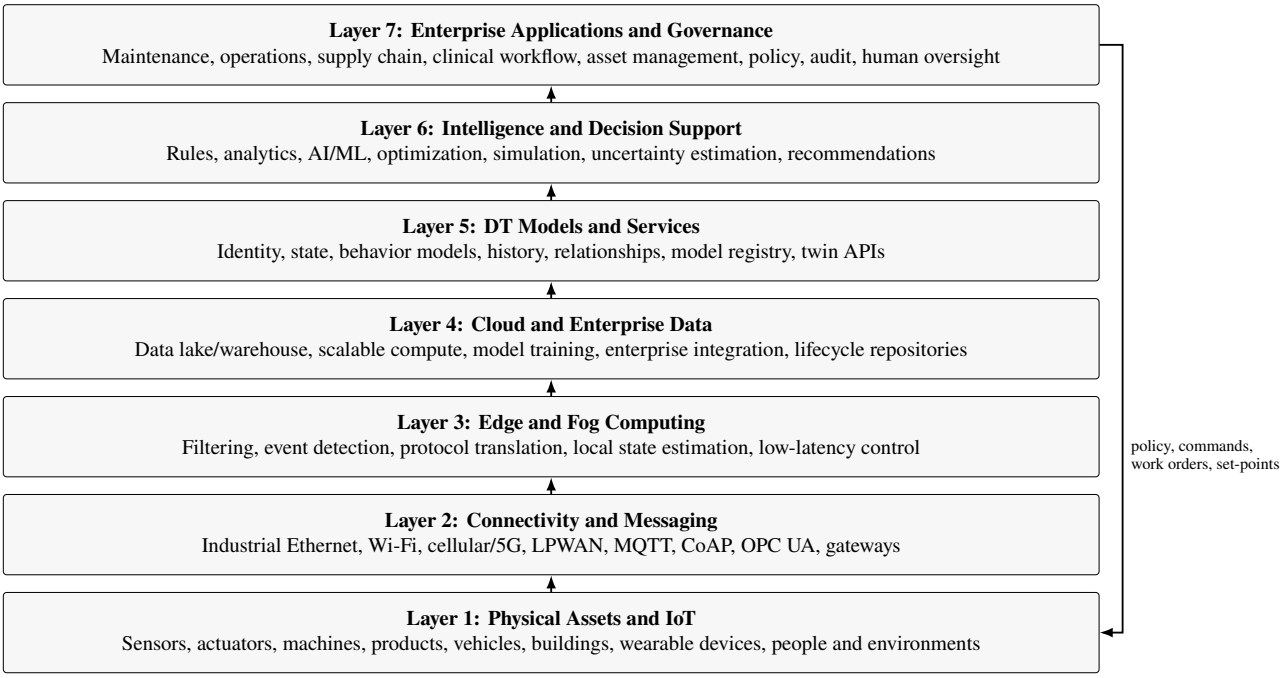


Figure 3: *Seven logical layers used in this review to organize IoT, DT, computing, decision, and enterprise functions.*

5.1. Layer 1: Physical Assets and IoT

Layer 1 contains the physical entity represented by the DT together with the devices used to observe or influence it. The physical device may be an individual asset, a person, a process, or a larger cyber-physical system. Its boundary needs to be clear when several entities are represented within the same application. In a factory, for example, separate twins may represent components, machines, production lines, products, or the plant as a whole. A smart-city implementation may instead connect models of buildings, roads, utilities, vehicles, and citizen-facing services. Human-centered applications introduce different constraints. In healthcare, the information that can be sensed, represented, and used is subject to consent, privacy, clinical accountability, and data-provenance requirements. The connection between the physical device and the DT begins with sensing and, where required, actuation. At this layer, observations also need to remain associated with device identity, measurement time, calibration information, and basic device-health status. Problems in these inputs may affect the state estimated at later layers. A miscalibrated sensor, for example, may provide measurements that lead to an inaccurate representation of the physical condition. A similar problem occurs when data are associated with the wrong

asset identifier. An enterprise DT implementation therefore requires an asset registry that provides a defined mapping between physical assets, sensing devices, and their digital identifiers.

5.2. Layer 2: Connectivity and Messaging

Layer 2 carries measurements, events, and control messages between physical devices and computing resources. It includes wired and wireless communication links, messaging protocols, gateways, and routing functions. The communication technology depends on the deployment environment and its timing, range, energy, and reliability requirements. Industrial systems may require deterministic communication, whereas mobile applications commonly depend on cellular or other wireless links. Resource-constrained IoT devices may instead use communication protocols designed for low-power operation. Higher layers should not have to interpret every device-specific transport or protocol directly. Gateways and protocol adapters can translate field-level communication into a common data representation before the information is passed to edge, cloud, or DT services. This separation also allows the communication technology to change without requiring corresponding changes in the DT model. Timing information is important because the DT must determine when an observation was produced and whether it still represents the current physical state. Delivery behavior must also be known. Depending on the communication protocol, messages may be lost, duplicated, delayed, or received in a different order from that in which they were transmitted. Network-oriented DT papers therefore consider communication delay and delivery behavior when evaluating synchronization and time-sensitive twin services [15, 22]. Stale or delayed observations can reduce the reliability of decisions that depend on the current state of the physical system.

5.3. Layer 3: Edge and Fog Computing

Layer 3 places selected computation close to the physical system. It can process IoT data before the information is transferred to cloud platforms or higher-level DT services. These include filtering, aggregation, event detection, protocol translation, local caches, short-horizon prediction, and control functions that cannot wait for a distant service. Fog/edge architectures also reduce upstream traffic by transmitting events or summaries instead of every raw sample [29]. For a DT, the edge can maintain a local shadow of state when cloud connectivity is interrupted. It can also apply a safety rule before accepting a control command. This makes the edge a useful enforcement point between enterprise-level optimization and physical actuation. The layer should not be treated as a miniature cloud without a defined purpose. Functions should be placed at the edge because their latency, data volume, privacy, or resilience requirement justifies the placement.

5.4. Layer 4: Cloud and enterprise data

Layer 4 provides scalable storage, compute, historical data, model training, and integration with enterprise information systems. It often contains time-series databases, object stores, data lakes, data warehouses, stream processors, and enterprise APIs.

5.5. Layer 5: DT Models and Services

Layer 5 maintains the digital representation of the physical entity. Its functions include twin identity, current state, relationships, model management, history, and service interfaces. Depending on the application, the representation may use physics-based models, discrete-event models, geometry, knowledge graphs, statistical models, or combinations of these approaches. DT technologies papers describe several model forms across different application domains [11, 13, 32]. Applications require a stable interface to twin information even when the internal model or data-processing method changes. A maintenance application, for example, may obtain an asset-health estimate or predicted failure interval through a defined twin service. A stable interface separates the application from details such as the preprocessing method used for the underlying measurements. This reduces coupling between application logic and a particular model implementation. The output can also retain references to the model version and input data used to produce it. These records provide a basis for tracing the result later in the twin lifecycle. Layer 5 also represents relationships among physical and logical entities. In a production line, machines may be associated with individual process stages. A building model may link zones with the equipment that serves them, while a supply-chain DT may connect facilities, inventory, and shipments. These relationships are needed when composition or dependency affects the represented state or the interpretation of a decision.

5.6. Layer 6: Intelligence and Decision Support

Layer 6 uses the state maintained by the DT for analysis and decision making. Its functions may involve rules, AI/ML models, optimization, simulation, forecasting, or uncertainty estimation. The output depends on the application and may take the form of a state estimate, forecast, recommendation, schedule, or control decision. The responsibilities of Layers 5 and 6 are separated in this architecture. Layer 5 maintains the identity, state, history, and relationships of the twin. Layer 6 uses that information to interpret current conditions or select an action. This separation reduces coupling between the maintained twin representation and the decision method. A decision model can then be evaluated, retrained, replaced, or disabled without redefining the identity or historical record of the twin. Decision authority is represented explicitly in the architecture. For this review, three modes are used:

- *advisory*, where information is presented to a user
- *supervised*, where a person approves the proposed action
- *autonomous*, where the system acts within a defined policy envelope.

The choice among these modes depends on the operational context and the consequences of an incorrect action. Some actions may be executed automatically within specified limits. Clinical or safety-related industrial applications, for example, may instead be configured to require human approval or an independent safety mechanism.

5.7. Layer 7: Enterprise Applications and Governance

Layer 7 connects DT outputs with the enterprise processes in which they are used. These may include maintenance and production planning, energy management, logistics, quality control, clinical workflows, asset investment, and compliance activities. Governance at this layer assigns authority over DT information and actions, including who may access or modify information, who may approve a proposed action, and who may act on the resulting output. Layer 7 also covers ownership and information-retention rules. Audit requirements, model-approval procedures, access rights, and escalation paths provide the organizational context needed to manage how DT information is used. These controls connect technical output with the organizational process in which it is used. A predictive maintenance model, for example, may provide an accurate estimate of failure risk, but an intervention still depends on maintenance capacity, production schedules, and the availability of spare parts. In this review, enterprise transformation is therefore associated with the point at which DT information is incorporated into an operational decision or workflow. Table 2 summarizes the responsibilities of the seven layers together with representative failure modes.

Table 2: *Responsibilities and representative failure modes across the seven-layer architecture.*

Layer	Primary responsibility	Representative technologies / artifacts	Typical failure mode
1	Observe and influence the physical entity	Sensors, actuators, PLCs, wearables, asset identifiers	Calibration error, missing context, ambiguous identity, device failure
2	Move data and commands	Industrial networks, Wi-Fi, 5G, LPWAN, MQTT, CoAP, OPC UA	Delay, loss, incompatible protocol, weak authentication
3	Local processing and resilience	Gateways, edge servers, local analytics, caches, state estimation	Resource exhaustion, inconsistent local/cloud state, unmanaged software
4	Scalable data and enterprise integration	Cloud compute, data lake, time-series database, ERP/EAM/MES integration	Siloed data, schema conflict, uncontrolled data duplication
5	Maintain twin state, models, and relationships	Physics/data-driven models, knowledge graphs, twin APIs, model registry	Model drift, stale state, unclear model validity, broken entity mapping
6	Analyze and decide	AI/ML, optimization, simulation, rules, uncertainty models	Poor generalization, opaque recommendation, unsafe automation
7	Apply and govern decisions	Workflows, dashboards, maintenance/clinical/operations systems, policy	No process ownership, weak accountability, decision not acted upon

6. Enabling Technologies and Integration Patterns

6.1. IoT Communication and Industrial Interfaces

IoT communication provides the data and command path between physical entities and the higher layers of the architecture. Different deployments can use MQTT, CoAP, OPC UA, cellular networks, private 5G, Wi-Fi, Ethernet, or other domain-specific communication technologies. The appropriate choice depends on requirements such as timing, range, energy consumption, reliability, device capability, and the operating environment. The communication layer must preserve the information required by the DT as data move between systems. This includes entity identity, timestamps, measurement units, quality indicators, and relevant security information. A protocol can transport a measurement successfully while still leaving the higher layers unable to interpret it correctly if this context is lost. Protocol translation may be required when devices and enterprise systems use different interfaces. A gateway or edge service can perform this translation before the data are passed to the DT. Keeping protocol-specific processing at a defined interface reduces the amount of device-specific logic that must be implemented inside twin services and enterprise applications.

6.2. Edge/Fog and Cloud Continuum

Edge, fog, and cloud resources provide different locations for executing DT functions. Papers of fog-based IoT and DT-assisted edge computing identify latency, bandwidth, mobility, resource availability, and privacy as factors that affect placement decisions [26, 29]. DT systems introduce an additional consistency problem when state or model information is maintained at more than one location. One possible arrangement is to maintain the state required for immediate local decisions at the edge while retaining a more complete historical record in the cloud. Local state can support short-response-time functions and continued operation during temporary connectivity loss. Cloud resources can support model training, analysis across multiple assets, and long-term lifecycle records. When DT state is replicated across edge and cloud resources, each replica needs to remain associated with the same physical entity and model version. Delayed or

incomplete synchronization may leave one location with an older representation of the physical state or with a different version of the model. Replicated twin state therefore requires explicit version information and a defined synchronization policy for propagating updates between execution locations.

6.3. Artificial Intelligence and Analytics

AI and analytical methods can operate at different stages of DT processing. Analysis close to the physical system may detect events or interpret sensor signals before further transmission. State-estimation methods use the available observations to infer information that is missing or cannot be measured directly. At the decision stage, analytical models may forecast future conditions, detect anomalies, recommend actions, or optimize resource use. Jan et al. [30] identify data quality, explainability, integration with existing systems, and model maintenance as deployment concerns for AI in Industry 4.0 applications. These issues become particularly relevant when analytical outputs are used to guide operational decisions. Uncertainty information is relevant when an incorrect prediction can affect an important decision. Instead of returning only a point estimate, a model can provide information about confidence, prediction intervals, or the conditions under which the estimate is considered valid. This information can be used by Layer 6 when deciding whether an output can be applied automatically or should be reviewed by an operator.

6.4. Semantics, Knowledge Representation, and Standards

A DT requires information about the meaning and relationships of the data it maintains. This includes entity type, relationships with other entities, measurement units, operating constraints, and lifecycle state. DT standards papers report progress in definitions, manufacturing information models, interfaces, and representation mechanisms [20, 21]. Some interoperability requirements nevertheless remain specific to the application domain. Knowledge graphs and other semantic representations can be used to describe relationships among assets, processes, documents, and observations. These relationships allow applications to query the DT using operational concepts rather than only raw device identifiers or sensor topics. For example, an application may need to identify all pumps associated with a production line or all building zones affected by a particular air-handling unit. Representing these relationships in the shared model avoids defining them independently in each application.

6.5. Cybersecurity and Privacy Technologies

Within the architecture considered in this review, security extends across the full information and decision path. Device identity and secure startup mechanisms protect access to sensing and actuation functions at the physical side. Communication between components requires authentication and protection against unauthorized modification or disclosure. Edge and cloud platforms add controls such as access management, software maintenance, isolation, and logging. Twin services apply authorization to specific entities and operations, while decision functions require protection of the models and outputs used to select an action. These controls serve different purposes and are most effective when used together. Intrusion-detection methods can identify suspicious activity, whereas identity and access mechanisms define which devices, services, or users are permitted to participate in the system. Tamper-evident records can protect selected logs, but the physical correctness of the original sensor measurement has to be established through other means. DT cybersecurity papers therefore describe security concerns across communication, identity, data, models, and control functions [16, 17].

6.6. Five Recurring Integration Patterns

We grouped the reviewed papers into five integration patterns based on how DT information is used between physical observation and operational action. The categories range from systems that mainly report current state to systems in which a decision is returned to a workflow or controller. The five patterns describe different levels of decision involvement but they are not intended as mandatory stages of technical maturity. Figure 4 summarizes the classification.

The five patterns classify DT systems according to what the twin actually does within the decision process. Monitoring or diagnostic systems can provide operational value without progressing to automated control. A system is treated as closed-loop only when a return path exists and the authority to act on DT output has been defined.

7. Application Domains and Transformation Outcomes

The reviewed papers span manufacturing, healthcare, smart cities, buildings, agriculture, logistics, and energy. The same seven logical layers can be used to compare these domains, although the physical entity being represented, the required update rate, the type of model, and the level of decision authority vary from one application to another. Table 3 provides a cross-domain comparison.

7.1. Manufacturing and maintenance

Manufacturing remains the clearest example of a DT because the physical assets, sensors, engineering models, and operational decisions can often be linked directly. Tao and Zhang described a DT shop-floor in which physical resources,

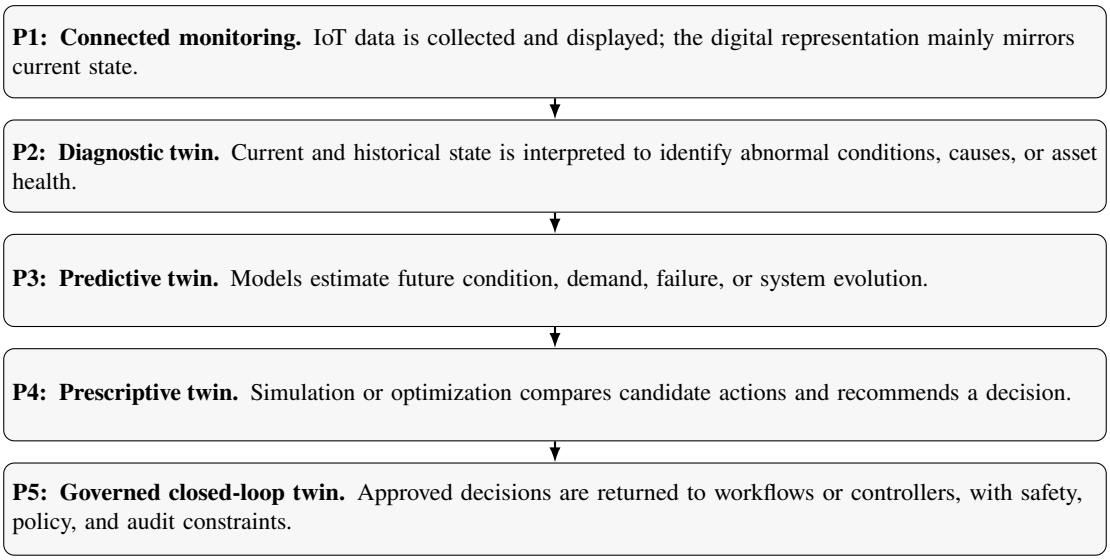


Figure 4: *Five integration patterns identified across the reviewed IoT-DT applications.*

Table 3: *Cross-domain comparison of IoT-DT applications.*

Domain	Physical device	Typical data	Decision function	Dominant constraint
Manufacturing	Machine, line, product, process	Condition signals, PLC states, quality, production history	Maintenance, quality, scheduling, process optimization	Legacy integration, timing, model validity
Healthcare	Patient, organ, care process, facility	Clinical observations, imaging, wearable and operational data	Risk assessment, personalized planning, capacity/flow support	Privacy, clinical validation, accountability
Smart city / infrastructure	Building, road, utility, district, urban system	Mobility, environment, asset, service and citizen data	Planning, operations, incident response, service optimization	Scale, heterogeneous ownership, interoperability
Energy / buildings	Equipment, building system, plant, grid asset	Metering, environmental, occupancy, equipment state	Energy optimization, fault detection, predictive maintenance	Control safety, temporal resolution, cross-system integration
Agriculture	Field, crop, irrigation system, farm asset	Soil, weather, imagery, water use, machine state	Irrigation, crop management, resource optimization	Sparse connectivity, environmental uncertainty
Supply chain / logistics	Shipment, warehouse, fleet, fulfillment system	Location, inventory, orders, equipment and process events	Routing, inventory, fulfillment, disruption response	Cross-organization data sharing and trust

virtual models, data, and services interact to support smart manufacturing [7]. Later reviews show that DT implementations cover design, production, maintenance, quality, and lifecycle services [10, 19]. Predictive and opportunistic maintenance shows the enterprise transformation path. A machine twin can use condition measurements to estimate the health of equipment and provide information for maintenance planning. Assad Neto et al. [33] applied a DT-based decision-support method to opportunistic preventive maintenance scheduling. In this case, the estimated equipment condition is considered together with available production opportunities. Maintenance can then be scheduled at a time that reduces interference with normal production. Manufacturing systems also illustrate the integration problems created by long-lived equipment and newer digital platforms. Machines may use proprietary interfaces, while engineering models, maintenance records, and production data may reside in separate systems. A DT implementation therefore has to connect operational technology, information technology, and engineering information even when these sources were not designed around a common data model. Standards, shared identifiers, and digital-thread mechanisms can reduce the amount of system-specific integration required over the asset lifecycle.

7.2. Healthcare

Healthcare DTs may represent organs, individual patients, or healthcare delivery processes. Elkefi and Asan reviewed DT applications for healthcare-system management and described approaches that combine operational and patient-related information [34]. Coorey et al. examined the health-DT concept for cardiovascular disease and discussed the need to combine patient data, computational models, and clinical interpretation [35]. Human-centered DTs introduce requirements that differ from those of equipment-oriented twins. Patient data are subject to consent, privacy, fairness, clinical validation, and accountability requirements. The interpretation of a model output also remains part of the clinical process. A predictive twin may estimate risk or simulate a possible intervention, but the result has to be considered together with clinical information that may not be represented completely by the model. Within the integration patterns defined in Section 6, many healthcare applications are closer to P3 or P4 because they support prediction or decision analysis. Moving such a system to P5 requires a clearly defined decision authority, validation procedure, and safety mechanism before model output can influence an intervention automatically.

7.3. Smart Cities and Public Infrastructure

Smart-city DTs combine information from infrastructure, transport, environmental sensing, buildings, utilities, and public services. White et al. presented a smart-city DT approach that incorporated citizen feedback [36]. Li et al. discussed city-scale DT concepts and their use in urban systems [37]. Recent papers also describe DT applications in urban planning, transportation, environmental monitoring, and infrastructure management [23]. At city scale, DT information is often distributed across several organizations. Transport agencies, utilities, building owners, public authorities, and private operators may each manage different assets and data sources. Connecting these sources requires access policies, common identifiers, and agreed meanings for the exchanged information. A city-level implementation may therefore be formed from several domain-specific twins that exchange selected information through defined interfaces. The required update frequency also depends on the urban function being supported. Traffic management may rely on frequently refreshed observations because road conditions can change within minutes. Land-use planning generally works with information that changes much more slowly. Separate but connected domain models allow each function to use an appropriate synchronization rate and level of model detail.

7.4. Energy Systems and Smart Buildings

DTs are used in energy and building systems to monitor operating conditions, forecast demand or performance, identify faults, and support control decisions. A building twin may combine building information models (BIM) with IoT measurements, occupancy information, and data from energy-management systems. Schweigkofler et al. examined DT-based energy management through the integration of IoT and BIM [38]. Renewable-energy research also considers DTs for resilience, maintenance, and sustainability across energy assets [39]. BIM and engineering models primarily describe design information, whereas IoT measurements describe the actual operating condition of a building or energy asset. The DT has to associate these two forms of information when the physical system changes over time. Closed-loop energy control introduces an additional requirement because optimization decisions can affect occupant comfort and equipment operation. Policy constraints, operating limits, and fallback behavior therefore belong in the decision and governance layers of the architecture.

7.5. Agriculture

Agricultural DT applications combine distributed sensing with processes that are strongly affected by environmental conditions. Soil and weather measurements can be combined with imagery, machinery records, and crop models to support irrigation and other farm-management decisions. Cesco et al. [40] examined the relationship between smart agriculture, DTs, and sustainability. Alves et al. [41] presented a smart-farming DT in which irrigation decisions were supported with the aim of reducing water use. Agricultural DTs may have to operate with limited sensing coverage or periods of weak connectivity. Local processing at the edge can keep selected functions available when communication with remote services is interrupted, with updated information exchanged after the connection is restored. The models also have to account for variation in weather, soil conditions, and biological processes. These sources of variation affect both the confidence of the estimated state and the amount of computing and communication that can reasonably be assigned to the DT.

7.6. Supply Chains, Warehouses, and Logistics

A supply-chain DT can represent facilities, inventory, transportation, demand, and operational events within the same connected system. Wang et al. [42] described a DT-driven supply-chain approach in which data and models support operational visibility and decision making. Warehouse applications include facility operation, resource coordination, and process simulation [43]. Ashrafian and Pedersen [44] presented a DT implementation for a large automated order-picking and fulfillment system. In these applications, information from one part of the supply chain can affect decisions elsewhere. A disruption in transportation, for example, may influence production schedules, inventory, procurement, or customer commitments. Integrating such information becomes more difficult when data are controlled by different organizations. Access rules and controlled data-sharing mechanisms are therefore needed, together with provenance information that records where decision-relevant data originated and under what conditions they were supplied.

7.7. Cross-Domain Interpretation

The application domains differ in their physical entities, operating environments, and decision requirements, but several common characteristics appear across the reviewed papers. IoT measurements need to be linked with a clearly identified entity and an appropriate model before they can be interpreted in a DT. The required synchronization rate and model detail depend on the decision being supported, so greater model complexity is not always necessary. Additional requirements appear when DT output is connected to operational action, particularly in relation to ownership, safety, access control, audit, and integration with existing processes. These observations provide a basis for evaluating enterprise transformation beyond the number of connected devices or DT instances. Relevant outcomes include changes in decision time, coordination between organizational functions, management of uncertainty, and the ability to support new or modified services while satisfying operational and governance requirements.

8. Cross-Domain Research Challenges

8.1. Semantic Interoperability

Interoperability occurs at several interfaces. Devices exchange measurements, software components exchange messages, and enterprise applications interpret the meaning of those messages. Protocol adapters and APIs can address part of the communication and syntactic problem. Semantic interoperability remains more difficult when systems use different identifiers, units, hierarchies, or interpretations for the same physical entity or property. Existing standards reduce some of these differences, although no single standard covers all of the enterprise domains considered in this review [20, 21]. An open research problem is how to connect domain-specific semantic models without requiring every organization to adopt one global ontology. Semantic version information is also relevant because entity definitions, units, and relationships can change across product generations and lifecycle stages.

8.2. Synchronization, Fidelity, and Validation

Errors in a DT can originate from several sources. Sensor measurements may be delayed or biased, the selected model may no longer represent the current operating condition, or the mapping between a physical entity and its digital instance may become incorrect. Production DTs reviews identify data integration, fidelity, validation, and synchronization among the recurring implementation challenges [19]. One research direction is to make the condition of the twin itself observable. State estimates can be accompanied by information about data age, source, measurement quality, uncertainty, and model version. These attributes would allow an application to determine how much confidence to place in a particular digital state. Validation may also need to be repeated during operation rather than performed only when a model is commissioned. Equipment wear, software changes, maintenance actions, or changes in operating conditions can alter the relationship between the model and the physical process. Methods for detecting and managing this loss of validity remain important for long-lived enterprise DTs.

8.3. Scalable Computation and Data Placement

Enterprise DTs may generate large data volumes and may also depend on computationally expensive models. Edge and cloud resources provide several execution locations, but placement becomes more difficult when multiple twins compete for processing, storage, and communication resources. Network- and edge-oriented DT research has begun to study computation offloading and resource orchestration [15, 26]. Enterprise deployments add requirements related to data ownership, model criticality, and operational continuity. A placement decision can therefore involve several objectives at the same time. These may include response-time limits, privacy constraints, computing cost, energy consumption, and resilience requirements. The cost of transferring or reconstructing twin state also becomes relevant when an edge service is migrated between computing nodes. Resource placement for DT functions may involve several objectives at the same time. Depending on the application, these can include response time, privacy, computing cost, energy use, resilience, and the cost of moving or reconstructing twin state.

8.4. Cybersecurity, Privacy, and Trust

Security problems can occur at several points in a DT system, from operational technology and communication links to cloud services, models, and enterprise applications. A failure in one part of this path may affect the information available to another component or the decision produced from it. DT cybersecurity reviews identify threats involving data integrity, authentication, privacy, communication, models, and control functions [16–18]. For operational decisions, this review separates two integrity concerns. *Semantic integrity* refers to whether data correctly describe the physical entity, state, or event to which they are assigned. Authorization establishes that a source is permitted to provide information, but the information itself may still be outdated, incorrect, or manipulated. *Decision integrity* concerns whether an action can be traced to the model or rule that produced it, the policy in effect at that time, and the approval status of the decision mechanism. Provenance and audit records can preserve this history. Model signing can help verify the integrity and origin of approved model artifacts, while access control limits who may modify or use them. Privacy becomes a separate concern when a DT represents a person or information associated with an individual. Work on Human DTs distinguishes human-centered models from conventional asset twins because the represented information can relate directly to a person [45]. Removing direct identifiers may still leave behavioral or contextual information that is sensitive. At the enterprise layer, controls can limit data collection to the stated purpose and define how access, retention, and consent are managed.

8.5. Lifecycle Management and Model Governance

The physical entity represented by a DT may remain in operation longer than the hardware and software used to maintain its digital representation. During that lifetime, sensors may be replaced and the represented asset may be reconfigured. Processing pipelines and enterprise applications may also change, while analytical models may be retrained as new data become available. These changes create a traceability problem when an earlier decision has to be examined later. It may be necessary to determine which sensor configuration, data source, model version, and software state were active when the

output was produced. Model registries, data-lineage records, configuration histories, and versioned interfaces can retain this information across the DT lifecycle. Modification rights are another part of lifecycle governance. A data engineer may change a processing pipeline, while an equipment engineer may revise a physics-based model. A predictor may later be retrained by a data scientist. The review and approval required for these changes can differ according to their effect on the DT and the risk of the application. For safety-related or regulated processes, role-specific change control and configuration records make it possible to identify who changed the system and which configuration was active when a decision was made.

8.6. Legacy Systems and Technical Debt

Enterprise environments may contain equipment and software introduced at different times and for different operational purposes. Data may be distributed across databases, spreadsheets, proprietary applications, and interfaces developed for individual systems. These existing dependencies remain part of the integration problem when a DT is introduced. If every connection is implemented separately, the DT can add another set of system-specific interfaces to the existing environment. Incremental encapsulation provides one option for handling such systems. Adapters can expose defined interfaces from existing applications or equipment, while common identifiers and semantic descriptions are introduced at higher layers. This allows parts of the DT architecture to be introduced gradually while existing systems continue to operate. The adapters then become software components in their own right and need version control, monitoring, security updates, and maintenance.

8.7. Evidence of Enterprise Value

Technical evaluation and enterprise evaluation answer different questions. Measures such as prediction accuracy, response time, energy consumption, and simulation error describe how a DT component or model performs. Enterprise evaluation concerns what changes in the process where that component is used. Relevant outcomes may include decision time, downtime, intervention frequency, resource use, coordination between organizational functions, or service performance. These metrics are necessary, but they do not show whether the organization changed its process. Digital transformation research warns against reducing transformation to technology adoption [1–3]. Enterprise DT evaluations should therefore report at least one process-level outcome in addition to technical metrics. Examples include maintenance lead time, unplanned downtime, energy cost, resource utilization, decision delay, service level, water use, or recovery time after a disruption. The causal attribution should be conservative. A before/after improvement cannot automatically be assigned to the DT if staffing, policy, or operating conditions changed at the same time. Table 4 maps the main challenges to the architecture interfaces at which they become visible.

Table 4: *Challenge-to-interface mapping for enterprise DT systems.*

Challenge	Primary interface	Failure if unmanaged	Required control / research direction
Semantic interoperability	Layers 1-5 and 5-7	Data can be transferred but not interpreted consistently	Shared identifiers, semantic models, mappings, versioned ontologies
Synchronization and fidelity	Layers 1-5	Twin state does not represent the current physical condition	Age/quality metadata, adaptive synchronization, continuous validation
Scalable placement	Layers 2-4-6	Excessive latency, bandwidth use, or cloud dependence	Edge/cloud orchestration with state migration and policy constraints
Cybersecurity and privacy	All layers	False state, unauthorized access, unsafe control, personal-data exposure	Identity, least privilege, provenance, model integrity, privacy-by-design
Lifecycle governance	Layers 4-7	Decisions cannot be reproduced after model or configuration change	Version control, model registry, audit, approval workflows
Legacy integration	Layers 1-4	Fragile point-to-point integration and duplicated semantics	Managed adapters, API contracts, incremental modernization
Value measurement	Layer 7	Technical success without process improvement	Process-level KPIs, controlled evaluation, explicit decision ownership

9. Research Agenda for Enterprise IoT-DT Systems

The preceding challenges suggest a research agenda that is more specific than calling for "better interoperability" or "more secure twins." Seven directions are outlined below. Each direction is tied to a failure mode observed across more than one application domain.

9.1. Composable and federated twins

Large enterprises are unlikely to operate one monolithic twin. They will operate many twins owned by different teams and vendors. A remaining research problem is how independently developed twins can be composed into larger systems. An asset twin may need to participate in a production-line twin, a building twin in a campus model, or a warehouse twin in a supply-chain representation. Such composition depends on shared identity, relationship semantics, interface contracts, and mechanisms for resolving inconsistent state or conflicting updates. Federation becomes necessary when DTs span several organizations or administrative domains. Smart-city and supply-chain systems, for example, may involve data and assets

controlled by different participants. A federated architecture can allow each participant to expose the information required for a shared decision while retaining control over data that should remain local. This creates both distributed-systems and governance requirements.

9.2. Edge-Native and Resource-Aware Twins

The placement of DT functions across edge and cloud resources remains an open design problem. State estimation, local models, safety functions, and time-sensitive decisions may need to execute close to the physical process, whereas model training, historical analysis, and computationally intensive simulation can use cloud resources. Network-oriented research provides a basis for resource-aware computation offloading [26]. Enterprise DTs add further requirements involving state consistency, software assurance, and fail-safe operation. One candidate design is a *minimum viable twin* that remains available at the edge during a loss of cloud connectivity. Such a twin would maintain the state, operating constraints, and safety information required for local continuity. More computationally intensive simulation, training, and cross-asset analysis could remain in the cloud. An unresolved issue is how the local and remote states should be reconciled after connectivity is restored, particularly when both copies have changed during the interruption.

9.3. Uncertainty-Aware and Self-Assessing Twins

DT state and model outputs can contain several forms of uncertainty. These include sensor quality, missing observations, state-estimation error, prediction uncertainty, and uncertainty in simulation results. Recording this information together with the estimated state would allow the decision layer to distinguish between outputs with different levels of confidence. Twin-health monitoring provides a related research problem. Data freshness, model validity, connectivity, and configuration consistency can be treated as indicators of whether the current digital representation remains suitable for a particular decision. A self-evaluation mechanism could use these indicators to adjust decision authority when confidence in the twin deteriorates. An autonomous function, for example, could shift to supervised or advisory operation when the reliability of its state information or the validity of its model falls outside defined limits.

9.4. Secure Twin Data Spaces and Decision Provenance

Protecting individual messages is one part of securing a DT system. When DT output is used to influence an operational action, it is also useful to retain evidence of how that action was produced. A decision record could link the physical observations used at the time, the data transformations applied to them, the resulting twin state, the model version, the applicable policy, and the actor responsible for approval. Such a record would provide a basis for reconstructing the decision path after the action has occurred. This traceability is referred to here as decision provenance. Decision provenance can be implemented through signed logs, trusted execution environments, access-control mechanisms, tamper-evident storage, or combinations of these techniques. The research problem is to provide verifiable records without introducing excessive storage, communication, or processing overhead in high-rate DT applications. Privacy-preserving analytics and federated learning provide another possible direction when several organizations need to construct shared models without transferring all raw data to one location.

9.5. Explainable and Human-Governed Decision Support

The explanation associated with an AI-supported DT decision should reflect the operational task in which that decision is used. Feature attribution may provide useful model information, but operational users can require additional context. A maintenance engineer, for example, may need recent sensor behavior, operating conditions, related historical events, and the uncertainty associated with a recommendation. Clinical use may require evidence that can be interpreted within the relevant clinical context together with a clear description of the model's intended use. Human oversight also needs an explicit place in the decision process. Research is required to determine which actions can be automated within defined limits, which require approval, and under what conditions an automated function should return control to a person. Human DT research makes this distinction particularly important because the modeled entity and the person affected by the resulting decision may be the same individual [45].

9.6. Sustainable DTs

DT applications can be used to support energy or resource optimization, but maintaining the twin also consumes computational, storage, communication, and sensing resources. Sustainability evaluation should therefore include the resources required to operate the DT itself. Frequent synchronization, high-resolution data collection, and compute-intensive inference can reduce the net resource benefit of an otherwise effective optimization method. Research on renewable-energy DTs already links DT operation with resilience and sustainability objectives [39]. At enterprise scale, evaluation can also include the computing and communication footprint of the twin. Adaptive model fidelity and workload placement provide possible mechanisms for reducing that footprint when lower-cost operation does not violate safety or service requirements. Energy and carbon conditions can also be considered when selecting execution locations for workloads that are flexible in time or placement.

9.7. Enterprise-Level Evaluation and Governance Metrics

Evaluation at enterprise level requires measures beyond algorithmic accuracy or execution time. The effect of a DT can also be assessed through changes in information quality, decision time, intervention rate, downtime, resource use, or service outcomes. The appropriate measure depends on the process for which the DT was introduced. Governance can be evaluated separately. Possible indicators include the proportion of active models with current validation, the percentage of operational decisions with complete provenance, the number of unauthorized configuration changes detected, or the time required to withdraw a compromised model from service. Such measures connect technical model management with the governance processes defined at Layer 7. Figure 5 summarizes the research directions from connected physical assets to governed enterprise DT operation. The stages indicate increasing integration of state, decision, and governance functions but they are not intended as a maturity score. An application may remain at an earlier stage when additional automation is unnecessary or when the associated operational risk is unacceptable.

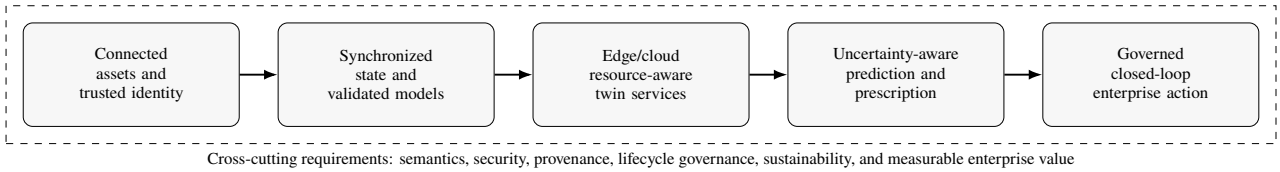


Figure 5: *Research directions from connected assets to governed enterprise DT systems.*

10. Limitations of the Review

The review has four limitations.

- The first concerns coverage. The paper uses a bounded structured analysis of 45 publications rather than an exhaustive bibliometric search. The selected papers cover architecture, enabling technologies, standards and security, and several application domains. The resulting analysis can therefore be used to compare recurring technical structures within this corpus, but it should not be used to estimate the prevalence of a topic across the complete DT literature.
- Terminology introduces a second limitation. Some publications use DT for systems that would be classified as digital shadows or connected digital models under other taxonomies. The terminology used by each source was preserved during extraction, while the information-flow distinction described in Section 2 was used when comparing architectures.
- The third limitation is that the available evidence also differs across application domains. Papers use different physical devices, validation procedures, operating conditions, and levels of decision authority. These differences limit direct comparison between, for example, industrial, healthcare, city-scale, and human-centered DT applications.
- Finally, enterprise outcomes are reported using different technical and organizational measures. The reviewed papers therefore do not provide a common quantitative basis for a meta-analysis of business or operational value.

11. Conclusion

IoT and DTs play related but different roles in enterprise systems. IoT infrastructure provides the connection to physical entities through sensing, identification, communication, and, where required, actuation. The DT uses information from that connection to maintain a representation of the corresponding asset or process, including its state, behavior, relationships, and associated models. Processing may take place at the edge, in the cloud, or across both environments according to the requirements of the application. Twin information can then be used by AI, simulation, optimization, or rule-based methods for analysis and decision support. In this review, the enterprise role of these technologies becomes most visible when their outputs are incorporated into an operational workflow or control process. The analysis resulted in seven logical layers: physical assets and IoT, connectivity, edge/fog computing, cloud and enterprise data, DT models and services, intelligence and decision support, and enterprise applications and governance. The reviewed papers were also grouped into five integration patterns: connected monitoring, diagnostic twins, predictive twins, prescriptive twins, and governed closed-loop twins. These patterns describe different uses of DT information and are intended as classification categories rather than mandatory stages of development. Systems that connect DT-generated decisions to an operational workflow or physical process bring additional concerns related to validation, cybersecurity, clear decision authority, governance, and audit. The reviewed applications cover manufacturing, healthcare, smart cities, buildings and energy, agriculture, and supply chains. Across these domains, recurring issues include aligning entity identities and semantics, maintaining sufficient consistency between digital and physical state, and determining where data and computation should be placed. Other concerns involve data and decision integrity, model changes over the system lifecycle, and integration with existing enterprise systems. These issues also influence one another. A change in synchronization policy may affect both communication load and the accuracy of the maintained state. Moving a model between edge and cloud resources may change privacy and resilience requirements. Similarly, when DT output is given greater authority to influence an operational action, validation and audit requirements may become more demanding. Taken together, the findings support an

observation-model-decision-action view of enterprise DT systems. Physical observations are associated with a maintained digital representation, that representation is used within a defined decision process, and the resulting information may be connected to an operational action under appropriate governance. Within the scope of this review, the usefulness of an enterprise DT therefore depends on how these parts work together across the complete information and decision path. A DT contributes to organizational processes when its information can be used for a defined decision and the resulting action can be traced, evaluated, and revised.

Future research should therefore focus on composable twins, edge-native execution, uncertainty-aware operation, secure decision provenance, human-governed AI, sustainability, and enterprise-level evaluation.

Author Declarations

Author Contributions (CRediT): Saira Shairi: Conceptualization, Methodology, Investigation, Data Curation, Formal Analysis. Saqib Dar: Visualization, Writing - Original Draft, Writing - Review & Editing. Both authors have read and approved the submitted version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: No new experimental dataset was generated.

Acknowledgments: No acknowledgments are reported.

Conflicts of Interest: The author declares no financial or personal conflict of interest.

Ethics Statement: Not applicable.

Declaration of Generative AI: During the preparation of this work, the authors used generative AI tools only for language editing and proofreading, where applicable. The authors reviewed and edited the content and take full responsibility for the published paper.

References

[1] G. Vial, "Understanding digital transformation: A review and a research agenda," *The Journal of Strategic Information Systems*, vol. 28, no. 2, pp. 118–144, 2019, doi: 10.1016/j.jsis.2019.01.003.

[2] P. C. Verhoef, T. Broekhuizen, Y. Bart, A. Bhattacharya, J. Q. Dong, N. Fabian, and M. Haenlein, "Digital transformation: A multidisciplinary reflection and research agenda," *Journal of Business Research*, vol. 122, pp. 889–901, 2021, doi: 10.1016/j.jbusres.2019.09.022.

[3] S. Nadkarni and R. Pruegl, "Digital transformation: a review, synthesis and opportunities for future research," *Management Review Quarterly*, vol. 71, pp. 233–341, 2021, doi: 10.1007/s11301-020-00185-7.

[4] Y. Lu, "Industry 4.0: A survey on technologies, applications and open research issues," *Journal of Industrial Information Integration*, vol. 6, pp. 1–10, 2017, doi: 10.1016/j.jii.2017.04.005.

[5] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, "Digital Twin in manufacturing: A categorical literature review and classification," *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1016–1022, 2018, doi: 10.1016/j.ifacol.2018.08.474.

[6] Q. Qi and F. Tao, "Digital Twin and big data towards smart manufacturing and Industry 4.0: 360 degree comparison," *IEEE Access*, vol. 6, pp. 3585–3593, 2018, doi: 10.1109/ACCESS.2018.2793265.

[7] F. Tao and M. Zhang, "Digital Twin shop-floor: A new shop-floor paradigm towards smart manufacturing," *IEEE Access*, vol. 5, pp. 20 418–20 427, 2017, doi: 10.1109/ACCESS.2017.2756069.

[8] A. Fuller, Z. Fan, C. Day, and C. Barlow, "Digital Twin: Enabling technologies, challenges and open research," *IEEE Access*, vol. 8, pp. 108 952–108 971, 2020, doi: 10.1109/ACCESS.2020.2998358.

[9] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, "Characterising the Digital Twin: A systematic literature review," *CIRP Journal of Manufacturing Science and Technology*, vol. 29, pp. 36–52, 2020, doi: 10.1016/j.cirpj.2020.02.002.

[10] 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, doi: 10.1016/j.jmsy.2020.06.017.

[11] Q. Qi, F. Tao, T. Hu, N. Anwer, A. Liu, Y. Wei, L. Wang, and A. Y. C. Nee, "Enabling technologies and tools for digital twin," *Journal of Manufacturing Systems*, vol. 58, pp. 3–21, 2021, doi: 10.1016/j.jmsy.2019.10.001.

[12] C. Qian, X. Liu, C. Ripley, M. Qian, F. Liang, and W. Yu, "Digital Twin—cyber replica of physical things: Architecture, applications and future research directions," *Future Internet*, vol. 14, no. 2, p. 64, 2022, doi: 10.3390/fi14020064.

[13] A. Sharma, E. Kosasih, J. Zhang, A. Brintrup, and A. Calinescu, "Digital Twins: State of the art theory and practice, challenges, and open research questions," *Journal of Industrial Information Integration*, vol. 30, p. 100383, 2022, doi: 10.1016/j.jii.2022.100383.

[14] X. Liu, D. Jiang, B. Tao, F. Xiang, G. Jiang, Y. Sun, J. Kong, and G. Li, "A systematic review of digital twin about physical entities, virtual models, twin data, and applications," *Advanced Engineering Informatics*, vol. 55, p. 101876, 2023, doi: 10.1016/j.aei.2023.101876.

[15] A. Hakiri, A. Gokhale, S. Ben Yahia, and N. Mellouli, "A comprehensive survey on digital twin for future networks and emerging Internet of Things industry," *Computer Networks*, vol. 244, p. 110350, 2024, doi: 10.1016/j.comnet.2024.110350.

[16] M. Homaei, O. Mogollon-Gutierrez, J. C. Sancho, M. Avila, and A. Caro, "A review of digital twins and their application in cybersecurity based on artificial intelligence," *Artificial Intelligence Review*, vol. 57, p. 201, 2024, doi: 10.1007/s10462-024-10805-3.

[17] S. R. Jeremiah, A. El Azzaoui, N. N. Xiong, and J. H. Park, "A comprehensive survey of digital twins: Applications, technologies and security challenges," *Journal of Systems Architecture*, vol. 151, p. 103120, 2024, doi: 10.1016/j.sysarc.2024.103120.

[18] M. El-Hajj, T. Itapelto, and T. Gebremariam, "Systematic literature review: Digital twins' role in enhancing security for Industry 4.0 applications," *Security and Privacy*, vol. 7, no. 5, p. e396, 2024, doi: 10.1002/spy2.396.

[19] S. Kerrouchi, E.-H. Aghezzaf, and J. Cottyn, "Production digital twin: a systematic literature review of challenges," *International Journal of Computer Integrated Manufacturing*, vol. 37, no. 10–11, pp. 1168–1193, 2024, doi: 10.1080/0951192X.2024.2314792.

[20] K. Wang, Y. Wang, Y. Li, X. Fan, S. Xiao, and L. Hu, "A review of the technology standards for enabling digital twin," *Digital Twin*, vol. 1, no. 1, p. 4, 2024, doi: 10.12688/digitaltwin.17549.2.

[21] G. Shao, "Manufacturing Digital Twin standards," in *2024 ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, 2024, pp. 1–8, doi: 10.1145/3652620.3688250.

[22] A. M. V. V. Sai, C. Wang, Z. Cai, and Y. Li, "Navigating the Digital Twin network landscape: A survey on architecture, applications, privacy and security," *High-Confidence Computing*, vol. 4, no. 4, p. 100269, 2024, doi: 10.1016/j.hcc.2024.100269.

[23] A. Huzzat, A. Anpalagan, A. S. Khwaja, I. Woungang, A. A. Alnoman, and A. S. Pillai, "A comprehensive review of Digital Twin technologies in smart cities," *Digital Engineering*, vol. 4, p. 100040, 2025, doi: 10.1016/j.dte.2025.100040.

[24] B. F. Mon, M. Hayajneh, N. Abu Ali, F. Ullah, H. Ullah, and S. Alkobaisi, "Digital Twins in the IIoT: Current practices and future directions toward Industry 5.0," *Computers, Materials & Continua*, vol. 83, no. 3, pp. 3675–3712, 2025, doi: 10.32604/cmc.2025.061411.

[25] P. T. J. C. Filho and O. Canciglieri Junior, "Digital Twin in Industry 4.0: Systematic review and content analysis and an architectural reference model," in *Procedia Computer Science*, vol. 253, 2025, pp. 2919–2928, doi: 10.1016/j.procs.2025.02.016.

[26] H. Tran-Dang and D.-S. Kim, "Digital Twin-empowered intelligent computation offloading for edge computing in the era of 5G and beyond: A state-of-the-art survey," *ICT Express*, vol. 11, no. 1, pp. 167–180, 2025, doi: 10.1016/j.icte.2025.01.002.

[27] S. Ahmetoglu, Z. Che Cob, and N. Ali, "A systematic review of Internet of Things adoption in organizations: Taxonomy, benefits, challenges and critical factors," *Applied Sciences*, vol. 12, no. 9, p. 4117, 2022, doi: 10.3390/app12094117.

[28] T. Zheng, M. Ardolino, A. Bacchetti, and M. Perona, "The applications of Industry 4.0 technologies in manufacturing context: a systematic literature review," *International Journal of Production Research*, vol. 59, no. 6, pp. 1922–1954, 2021, doi: 10.1080/00207543.2020.1824085.

[29] M. Burhan, H. Alam, A. Arsalan, R. A. Rehman, M. Anwar, M. Faheem, and M. W. Ashraf, "A comprehensive survey on the cooperation of fog computing paradigm-based IoT applications: Layered architecture, real-time security issues, and solutions," *IEEE Access*, vol. 11, pp. 73 303–73 329, 2023, doi: 10.1109/ACCESS.2023.3294479.

[30] Z. Jan, F. Ahamed, W. Mayer, N. Patel, G. Grossmann, M. Stumptner, and A. Kuusk, "Artificial intelligence for industry 4.0: Systematic review of applications, challenges, and opportunities," *Expert Systems with Applications*, vol. 216, p. 119456, 2023, doi: 10.1016/j.eswa.2022.119456.

[31] C. Daase, C. Haertel, A. Nahhas, M. Volk, H. Steigerwald, A. Ramesohl, B. Schneider, A. Zeier, and K. Turowski, "Following the digital thread – a cloud-based observation," in *Procedia Computer Science*, vol. 217, 2023, pp. 1867–1876, doi: 10.1016/j.procs.2022.12.387.

[32] J.-F. Yao, Y. Yang, X.-C. Wang, and X.-P. Zhang, "Systematic review of digital twin technology and applications," *Visual Computing for Industry, Biomedicine, and Art*, vol. 6, p. 10, 2023, doi: 10.1186/s42492-023-00137-4.

[33] A. Assad Neto, B. S. Carrijo, J. G. R. Brock, F. Deschamps, and E. P. de Lima, "Digital twin-driven decision support system for opportunistic preventive maintenance scheduling in manufacturing," in *Procedia Manufacturing*, vol. 55, 2021, pp. 439–446, doi: 10.1016/j.promfg.2021.10.060.

[34] S. Elkefi and O. Asan, "Digital Twins for managing health care systems: Rapid literature review," *Journal of Medical Internet Research*, vol. 24, no. 8, p. e37641, 2022, doi: 10.2196/37641.

[35] G. Coorey, G. A. Figtree, D. F. Fletcher, V. J. Snelson, S. T. Vernon, D. Winlaw, S. M. Grieve, A. McEwan, J. Y. H. Yang, P. Qian *et al.*, "The health digital twin to tackle cardiovascular disease—a review of an emerging interdisciplinary field," *npj Digital Medicine*, vol. 5, p. 126, 2022, doi: 10.1038/s41746-022-00640-7.

[36] G. White, A. Zink, L. Codeca, and S. Clarke, "A digital twin smart city for citizen feedback," *Cities*, vol. 110, p. 103064, 2021, doi: 10.1016/j.cities.2020.103064.

[37] D. Li, W. Yu, and Z. Shao, "Smart city based on digital twins," *Computational Urban Science*, vol. 1, p. 4, 2021, doi: 10.1007/s43762-021-00005-y.

[38] A. Schweigkofler, O. Braholli, S. Akro, D. Siegele, P. Penna, C. Marcher, L. Tagliabue, and D. Matt, "Digital Twin as energy management tool through IoT and BIM data integration," in *CLIMA 2022 Conference*, 2022, doi: 10.34641/clima.2022.46.

[39] W. A. Khan, A. Pakseresht, C. Chua, and A. Yavari, "Digital twin role for sustainable and resilient renewable power plants: A systematic literature review," *Sustainable Energy Technologies and Assessments*, vol. 75, p. 104197, 2025, doi: 10.1016/j.seta.2025.104197.

[40] S. Cesco, P. Sambo, M. Borin, B. Basso, G. Orzes, and F. Mazzetto, "Smart agriculture and digital twins: Applications and challenges in a vision of sustainability," *European Journal of Agronomy*, vol. 146, p. 126809, 2023, doi: 10.1016/j.eja.2023.126809.

[41] R. G. Alves, R. F. Maia, and F. Lima, "Development of a Digital Twin for smart farming: Irrigation management system for water saving," *Journal of Cleaner Production*, vol. 388, p. 135920, 2023, doi: 10.1016/j.jclepro.2023.135920.

[42] L. Wang, T. Deng, Z.-J. M. Shen, H. Hu, and Y. Qi, "Digital twin-driven smart supply chain," *Frontiers of Engineering Management*, vol. 9, no. 1, pp. 56–70, 2022, doi: 10.1007/s42524-021-0186-9.

[43] M.-J. Belanger, R. Pellerin, and S. Lamouri, "A literature review on Digital Twins in warehouses," in *Procedia Computer Science*, vol. 219, 2023, pp. 370–377, doi: 10.1016/j.procs.2023.01.302.

[44] A. Ashrafian and S. Pedersen, "Digital Twin for complex logistics systems: The case study of a large-scale automated order picking and fulfillment system," in *IFAC-PapersOnLine*, vol. 56, no. 2, 2023, pp. 11 056–11 061, doi: 10.1016/j.ifacol.2023.10.808.

[45] B. Gaffinet, J. Al Haj Ali, Y. Naudet, and H. Panetto, "Human Digital Twins: A systematic literature review and concept disambiguation for Industry 5.0," *Computers in Industry*, vol. 166, p. 104230, 2025, doi: 10.1016/j.compind.2024.104230.