Yang, Chao; Yu, Hao; Zheng, Yuan; Feng, Lei; Ala-Laurinaho, Riku; Tammi, Kari

A digital twin-driven industrial context-aware system : A case study of overhead crane operation

Published in:
Journal of Manufacturing Systems

DOI:
10.1016/j.jmsy.2024.12.006

Published: 01/02/2025

Document Version
Publisher's PDF, also known as Version of record

Please cite the original version:
Yang, C., Yu, H., Zheng, Y., Feng, L., Ala-Laurinaho, R., & Tammi, K. (2025). A digital twin-driven industrial context-aware system : A case study of overhead crane operation. *Journal of Manufacturing Systems*, *78*, 394-409. https://doi.org/10.1016/j.jmsy.2024.12.006

Journal of Manufacturing Systems 78 (2025) 394–409

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Technical paper

A digital twin-driven industrial context-aware system: A case study of overhead crane operation

Chao Yang [a,*], Hao Yu [b,*], Yuan Zheng [c], Lei Feng [d], Riku Ala-Laurinaho [a], Kari Tammi [a]

[a] Department of Energy and Mechanical Engineering, Aalto University, Puumiehenkuja 5, Espoo, 02150, Finland
[b] ICTFicial OY, Koivuviidantie 12, Espoo, 02130, Finland
[c] Department of Civil Engineering, Aalto University, Rakentajanaukio 4, Espoo, 02150, Finland
[d] Department of Machine Design, KTH Royal Institute of Technology, Brinellvägen 83, Stockholm, 10044, Sweden

A R T I C L E I N F O

Keywords:
Augmented reality
Context-aware system
Digital twin
Human-centered
Semantic technology

A B S T R A C T

With advancements in Information and Communication Technologies (ICT), traditional manufacturing industries are engaged in a digital transformation. This transformation enables the acquisition of vast amounts of data and information, enhancing decision-making capabilities. This, in turn, has raised the expectations of field operators who seek data and information management tailored to the dynamic working environment, thereby improving efficiency in their daily operations. However, there is a lack of a holistic approach to integrating diverse data sources, extracting valuable contextual information, and delivering real-time information to field operators. This paper addresses this gap by proposing an adaptive, interoperable, and user-centered Context-Aware System (CAS). Initially, the paper explores the challenges and requirements associated with CAS's current practices while proposing potential solutions. Furthermore, it introduces a system framework of CAS that integrates Digital Twin (DT) and semantic technologies. This framework includes three primary technical solutions: (1) Integrating DT to create a comprehensive digital representation of physical entities, enabling real-time data integration and synchronization; (2) Providing an ontology-based approach to model manufacturing context, facilitating knowledge representation and reasoning; (3) Developing a user-centered information delivery system leveraging Augmented Reality (AR) for context-aware visualization. The system architecture has been implemented and tested in a laboratory-scale industrial environment, focusing on crane operations within logistics scenarios. Lastly, three use cases are presented to demonstrate the system's practical applicability, showcasing its feasibility in furnishing informed contextual information to end-users within the dynamic manufacturing environment.

1. Introduction

Smart manufacturing [1] is transforming the industrial landscape through the use of data from interconnected devices and systems. This vast influx of data enables real-time insights into production processes, driving optimization and the transition from mass production to mass customization [2]. This shift necessitates adaptability and real-time decision-making, especially for factory operators managing unexpected events in dynamic environments such as small-batch customized production. These challenges demand advanced, real-time information distribution systems that can provide contextualized insights directly to field operators [3].

Context awareness is crucial for effectively utilizing data and information in changing settings [4], allowing the interpretation and adaptation of the surrounding environment. By understanding the context in which data is generated or information is utilized, Context-Aware System (CAS) enables users to make more informed decisions, adapt their behavior, and take relevant actions to the specific situation [5]. CAS can support patient monitoring and treatment in healthcare, providing real-time data to medical professionals for more accurate diagnoses and personalized care [6]. CAS can enhance the shopping experience in the retail sector by delivering tailored product recommendations and promotions based on a customer's location and preferences [7]. Despite the aforementioned advancements, deploying such systems for direct use by field operators in manufacturing environments presents significant challenges.

A key challenge for CAS in manufacturing is the complexity and variability of industrial processes, which involve numerous dynamic

* Corresponding authors.
E-mail address: chao.1.yang@aalto.fi (C. Yang).

https://doi.org/10.1016/j.jmsy.2024.12.006
Received 27 August 2024; Received in revised form 27 October 2024; Accepted 10 December 2024
Available online 26 December 2024

variables and workflows that are difficult to model and adapt to in real-time. Insufficient understanding of the physical environment further poses the challenge of adaptability, particularly in the face of increasingly complex and dynamic factory environments [8]. Furthermore, the integration of diverse data sources and formats while ensuring interoperability and scalability adds to the complexity of CAS. Manufacturing environments generate vast amounts of heterogeneous data from various sources, including sensors, machinery, and enterprise systems. Each of these sources may use different protocols, formats, and data structures, resulting in a lack of interoperability that disrupts collaboration across manufacturing systems, processes, information flow, organizational structures, and personnel [9]. Additionally, integrating CAS with human operators requires personalized, adaptive interfaces that present information in a user-centered way, ensuring the system enhances decision-making rather than overwhelming users. Thus, it is crucial to adopt a comprehensive approach for CAS in the manufacturing domain that prioritizes adaptability, interoperability, and user-centricity.

This paper addresses the aforementioned challenges in manufacturing operations by proposing a novel framework that integrates Digital Twin (DT) and CAS. Our framework leverages DT to collect and analyze real-time data from physical spaces, enabling advanced simulation, machine learning, and data-driven services. Real-time in this context refers to the ability to collect, analyze, and act on data from physical spaces almost instantaneously, with minimal latency, ensuring timely information delivery to end users. Central to the framework is a contextualization ontology that provides a structured knowledge base, facilitating efficient interpretation of complex relationships within the data. We also introduce a user-centered system that employs Augmented Reality (AR) to deliver customized information visualizations based on user preferences, such as user-defined dashboards. This AR-based interface enhances operator perception and interaction by overlaying relevant digital data onto the physical workspace, bridging the gap between the physical and digital realms. By fostering seamless communication and collaboration, this integrated approach enables stakeholders to gain deeper insights, mitigate risks, and improve overall operational efficiency in manufacturing environments.

The paper is organized as follows: Section 2 offers an overview of the relevant literature and background knowledge. In Section 3, the paper introduces the system architecture for the CAS. Section 4 outlines the system implementation that aligns with the proposed framework. Section 5 presents a case study of overhead crane operation within logistics scenarios. This is followed by an evaluation of quality attribute scenarios and a usability test. Following this, Section 7 engages in a discussion regarding the research's contributions, limitations, and potential avenues for future research. Finally, the paper concludes with a summary of the study. Abbreviations used throughout this article are listed in Table A.1 in the appendix.

The main contributions of this work are summarized as threefold:

- Integrating DT technologies with CAS to provide the platform with a holistic understanding of the manufacturing ecosystem.
- Establishing a contextualization ontological model, enabling seamless communication and data exchange.
- Creating an AR-based user-centered information delivery system for visualizing contextual information.

2. Literature review

This section first reviews the fundamental concepts of context and context-aware. Then, we analyze current CAS in the manufacturing domain and address challenges and limitations posed by existing industrial CAS. Furthermore, we explore potential technologies to solve these obstacles, including DT and semantic technologies.

2.1. Context

Numerous scholars have proposed definitions of context, aligning with their specific research domains [10,11]. One of the most widely cited is from Dey et al. [12], who define context as "any information that can be used to characterize the situation of an entity," where an entity includes persons, places, or objects relevant to an interaction. This definition emphasizes temporal, spatial, and relational factors. In the realm of semantic data modeling, context can also refer to "the circumstances in which something exists or occurs" [13], allowing for distinguishing individual entities and their relationships based on identified features.

Context can be identified and classified in various ways. Dey [14] distinguished between primary (e.g., time, identity, location, activity) and secondary contexts, the latter derived from the former. Similarly, Zimmermann et al. [15] categorized context into individuality, activity, location, time, and relations, focusing on spatio-temporal and relational attributes. More recent classifications have split contexts into external (e.g., location, physical states) and internal (e.g., user goals, business processes, emotional states) dimensions, with internal aspects captured via user interactions [16,17]. It is important to acknowledge that while existing fundamental categories may provide a foundational knowledge structure, their generic nature can be limiting when applied to the specific requirements of industrial environments. To address this, Zainol and Nakata [18] proposed extrinsic, interface, and intrinsic context categories, which were adapted for social robotics by Mahieu et al. [19]. In an industrial setting, Rosenberger et al. [20] introduced core contexts (user, environment, and system) alongside domain-specific contexts. The model proposed in this work integrates Zainol's and Rosenberger's classifications to better meet the needs of industrial applications.

2.2. Context-aware systems in manufacturing domain

Schilit and Theimer [21] first introduced the term context-aware to describe systems enabling interaction with nearby devices and services based on location as the core context element. Lucke et al. [22] expanded this to describe systems that adapt their behavior based on the operational context, enhancing system effectiveness and relevance. CAS generally refers to systems that use contextual information to provide relevant services or data [14,23]. In manufacturing, context awareness is increasingly important for improving operational visibility, production efficiency, worker safety, and overall productivity [16]. Alexopoulos et al. [3] proposed a context-aware manufacturing information system utilizing technologies like Near Field Communication (NFC), Radio-Frequency Identification (RFID), Resource Description Framework (RDF) to provide real-time information to operators. CAS has also been used for decision support in maintenance [24,25], estimating the duration of changeover-prone activities [26], and enhancing the adaptability of production systems [27]. In Human-Robot Collaboration (HRC), CAS improves safety by recognizing human poses to anticipate worker movements [28]. By providing a systematic understanding of the manufacturing system, CAS effectively bridges the gap associated with fragmented industrial information and limited expert knowledge, thereby empowering more informed decision-making.

The aforementioned CAS focused on offering distinct system frameworks, however, they lacked a holistic understanding of their physical counterparts. Furthermore, these systems exhibited deficiencies in establishing a comprehensive and formalized context model within the manufacturing domain. Modeling the context of systems in the manufacturing domain is necessary to enable interoperability and scalability [29]. As a result, the proposed architectures pose challenges for further information interchange and system integration due to issues related to semantic interoperability. Thus, there is a need for innovative solutions that prioritize mirroring the physical environments throughout the whole lifetime and establishing standardized approaches for modeling context in the manufacturing domain, while enhancing the effectiveness and usability of CAS in manufacturing.

2.3. Digital twin and context-aware system combination

In 2003, Michael Grieves first presented the concept of DT on product life-cycle management. DT refers to a virtual representation of the physical world throughout its lifecycle, providing advanced decision-making [30]. The framework of a DT comprises three interrelated layers [31,32]: (1) The physical layer encompasses physical assets, including components, actuators, and sensors, from which data is collected to capture the real-time state and behavior of the physical entities; (2) The information layer is related to data processing, data mapping, and data storage; (3) The virtual layer holds diverse digital formats that describe the physical object or system to predict behavior, optimize operations, and provide actionable insights. DTs have found extensive utility in smart manufacturing. By integrating real-time data from sensors and other sources, DTs provide a dynamic and accurate representation of the manufacturing process. This enables timely and informed decision-making, such as optimizing production schedules, predicting equipment failures, and improving product quality [33]. Incorporating DTs in manufacturing processes allows for the efficient processing of information generated by Cyber-Physical Systems (CPS). This facilitates the management of dynamic production flows and enhances flexibility for individualized manufacturing [34]. Real-time data acquisition and mapping within the workshop environment enable DTs to accurately reflect actual manufacturing processes and equipment conditions [35]. CAS requires dynamic operational adjustments in response to real-time information from complex and dynamic manufacturing environments. Integrating DTs provides a comprehensive and continuously updated virtual representation of physical assets and processes, thereby enhancing situational awareness and decision-making capabilities.

2.4. Semantic technologies

Semantic technologies provide advanced means for categorizing and processing data, as well as discovering relationships within varied data sets. Ontologies play a crucial role in semantic technology by providing a formal specification of a conceptualization relevant to a particular domain [36]. It acts as an approach that defines, captures, and standardizes information, enabling the seamless sharing and reuse of information, data, and domain knowledge [37]. The Web Ontology Language (OWL) is one of the most popular knowledge representation languages for authoring ontologies. OWL is a computational logic-based language, that enables computers and software applications to process, interpret, and manipulate information, facilitating data integration and automated reasoning. In CAS, it is imperative to establish an efficient context model [17]. Context modeling aims to capture and present contextual data, allowing for defining context types, attributes, and relationships within the context information [16]. Various approaches can be employed for context modeling, including key–value, markup scheme, graphical and object-oriented, logic-based, and semantic modeling [4]. The selection of a specific approach primarily depends on the particular application domain. In the context of industrial manufacturing, primary concerns are the issues of data heterogeneity and semantic interoperability, especially as new data sources are integrated and the production industry evolves [38]. Ontology, characterized by formalized knowledge representation, reusability, and reasoning capabilities, proves a well-suited solution for addressing semantic heterogeneity in industrial information systems [39]. The utilization of ontology also facilitates the development of machine-readable information models, fostering mutual comprehension among information systems within the industrial domain [40]. Consequently, this research adopts an ontology-based approach for context modeling to improve semantic interoperability.

2.5. Research motivation

In the manufacturing process, CAS can streamline the tasks of field operators by offering continuous monitoring of their actions and surroundings to provide contextual information and facilitate informed decision-making. Manufacturing environments generate vast amounts of data from diverse sources like sensors, Internet of Things (IoT) devices, and information systems. Despite advancements, several research gaps persist:

- The challenge of handling dynamic environments with constantly changing conditions.
- The integration of heterogeneous data sources to maintain data consistency and quality.
- The need for personalized, real-time information delivery tailored to individual user preferences.

To address these gaps, we propose a DT-driven CAS that overcomes these challenges. Additionally, the system leverages AR to provide user-centered information visualizations, catering to individual operator needs and preferences.

3. Architecture of digital twin-driven context-aware framework

The general architecture of the proposed CAS is illustrated in Fig. 1. The architecture comprises two key components: the DT Engine, responsible for handling digital counterparts of physical entities within the factory field; and the Context-aware Platform, which facilitates the management of contextual information.

3.1. Digital twin engine

The architectural design of the DT Engine follows a three-layer structure, consisting of the physical, information, and virtual layers, as described in Section 2.3. The DT Engine consists of three layers: (1) Physical Layer encompasses physical assets and industrial systems from which data is collected to capture the real-time state and behavior of the physical entities. (2) Information Layer is related to data and knowledge collection, transmission, processing, and integration. (3) Virtual Layer holds diverse digital models, e.g., AI models and 3D simulation models, that describe the physical object or system to predict behavior, optimize operations, and provide more insights.

The Physical Layer represents a complex and dynamic production environment, where the workforce employs internal and outsourced resources to carry out production tasks within the physical production unit, transforming the raw materials or semi-finished products into finished products [39]. Within the factory floor, three physical entities (e.g., agent, machine, and material) emerge as central to the production process, making it crucial to understand the basic elements that add value. Various IoT technologies are employed for real-time data acquisition from these entities with a focus on their states, conditions, and events. Concurrently, Operational Technology (OT) is leveraged to monitor and control physical processes, devices, and infrastructure, while Information Technology (IT) systems manage data and applications.

The Information Layer is responsible for acquiring and processing data to facilitate these DTs' development. To create a virtual model that accurately represents real-world entities or processes, DT relies on acquiring data about the entity or process. The Information Layer mainly encompasses four key processes: data collection, data transmission, data processing, and data integration. These processes are instrumental in gathering and incorporating data to enhance the effectiveness and accuracy of the DT's virtual representation. Data collection involves gathering real-time data from various sources in the manufacturing environment, including sensors, IoT devices, machinery, and control systems. The collected data is typically transmitted using industrial communication standards like Modbus, OPC Unified Architecture (OPC UA), and Message Queuing Telemetry Transport (MQTT). While non-time-critical data perception may involve gathering data from quality control systems, inventory management systems, and other relevant sources. Conventional protocols like Hypertext Transfer

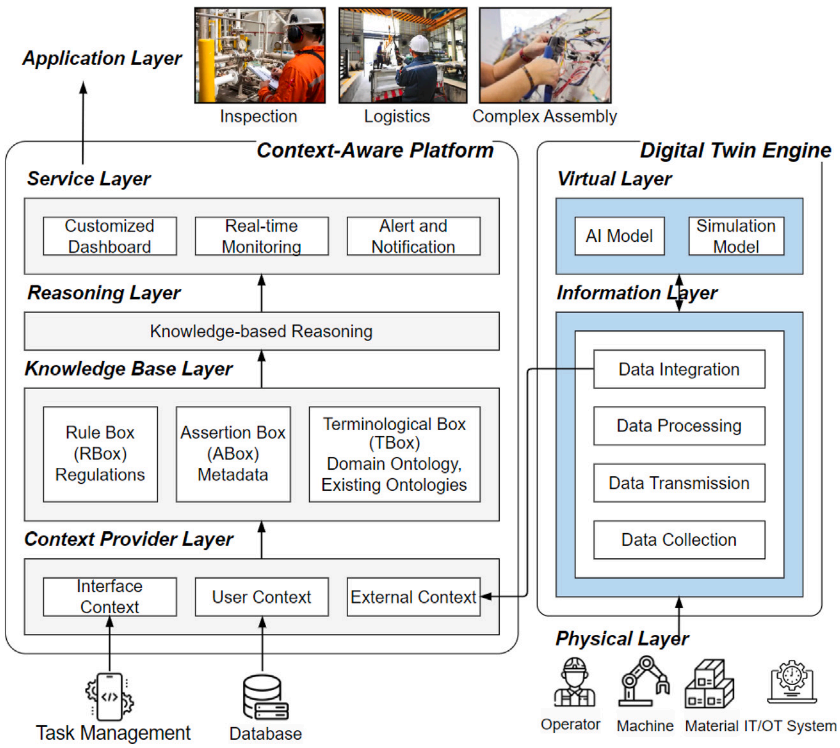

Fig. 1. General architecture of the DT-driven context-aware system.

Protocol (HTTP) can be employed to transmit this data. Data processing converts raw data into meaningful information, ensuring alignment with the predefined data format. The swiftness of data processing is imperative for timely responses to events, thereby enhancing operational efficiency. Finally, data integration involves combining processed data from various sources into a unified and coherent digital representation. This step ensures that the data is aligned and consistent, facilitating comprehensive analysis and decision-making. The Information Layer enables the dynamic adaptation and evolution of the DT over time, ensuring its continued relevance and effectiveness in addressing evolving system requirements and challenges.

The Virtual Layer in the DT Engine enables seamless feedback between the digital and physical spaces. This layer hosts 3D simulation models to visualize and predict the behavior of physical entities. For instance, real-time sensor data from the Physical Layer is processed within the Virtual Layer to detect safety risks, such as collisions. The results are then delivered back to the end users in the physical environment via the Context-aware Platform, as detailed in Section 5.5. This bidirectional interaction allows the system to not only mirror the physical environment but also take proactive actions, such as adjusting operations or providing operators with visual alerts. Additionally, AI models are leveraged to analyze vast data streams from the factory floor, including tasks like object recognition, as discussed in Section 5.4. The integration of semantic technologies further enhances CAS's capability to unify diverse data from simulations and AI models and deliver contextually relevant information (see Section 6.2).

3.2. Context-aware platform

The Context-aware Platform facilitates the acquisition of contextual information and supports context reasoning processes, which can be divided into four layers, including the Context Provider Layer, the Knowledge Base Layer, the Reasoning Layer, and the Service Layer. The Context Provider Layer manages the acquisition of three types of context information: external context, user context, and interface context, subsequently adding this information to the Knowledge Base Layer. The Knowledge Base Layer aggregates all the relevant context information into a formal context model, containing three essential components: Terminological Box (TBox), Assertion Box (ABox), and Rule Box (RBox). TBox describes formalized domain knowledge by defining classes and properties. ABox, on the other hand, represents instance-level information associated with TBox's ontologies. RBox defines logical relationships and infers new information based on the existing knowledge represented in the ontologies [41]. This paper introduces a contextualization ontology as a domain ontology within the TBox for managing contextual data in the factory environment. Existing ontologies are integrated into the platform and extended with this domain ontology which models the heterogeneous information from the factory environment. The existing ontologies include OWL-TIME [42], Building Topology Ontology (BOT) [43], Sensor, Observation, Sample, and Actuator (SOSA) ontology [44], Quantities, Units, Dimensions and Types (QUDT) [45], Industrial Production workflow ontology (InPro) [39], and GeoSPARQL [46], enabling the description of temporal, spatial, procedural, and sensor dimensions. The following section provides comprehensive details regarding the proposed domain ontological model. The ABox aggregates all the relevant contextual information from various sources into standardized metadata based on the domain ontology defined in the TBox. These information sources include IoT sensors, simulation environments, and AI models. By mapping sensor readings and other data points to the classes and properties defined in the ontology, ABox establishes a well-defined and interpretable representation of the manufacturing environment.

The Reasoning Layer is instrumental in deriving new insights and knowledge from the vast contextual information stored in the Knowledge Base Layer. Regulations and standards can be codified into executable rules within the Reasoning Layer. These rules facilitate efficient information retrieval and monitoring processes. For instance, the system can automatically identify potential safety hazards by comparing real-time sensor data from the shop floor with pre-defined safety regulations. The Reasoning Layer also plays a crucial role in ensuring that the right time for delivering accurate information is identified by continuously analyzing and interpreting the evolving context within the

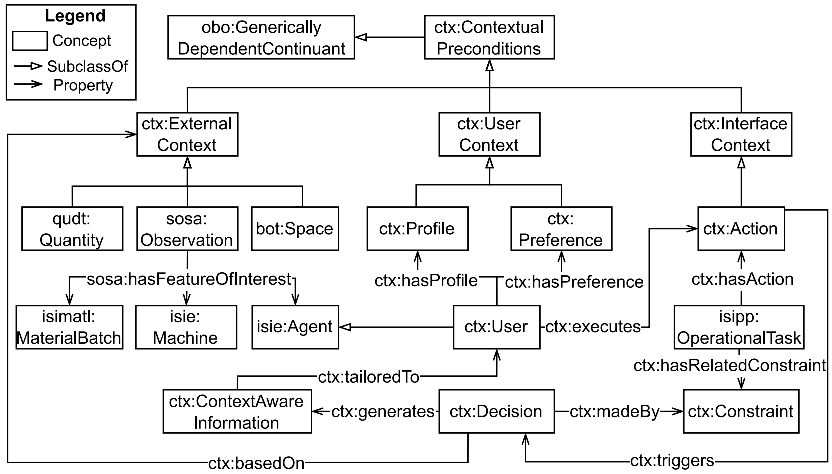

Fig. 2. Ontological model for contextualization.

manufacturing environment. This means the system not only processes data in real-time but also determines when specific information is most relevant and necessary based on the operational context or the user's current task. Finally, the insights and knowledge extracted by the Reasoning Layer empower the Service Layer to deliver abundant contextual services to various stakeholders within the manufacturing environment. These services leverage the reasoning capabilities to provide right-time operational support and facilitate proactive decision-making for human-centered industrial processes, including inspection, logistics, and complex assembly. The Service Layer ensures that the right information reaches the right person at the right moment, enhancing decision-making and reducing operational risks.

3.3. Ontological model for contextualization

Industrial environments generate a wealth of data that can be leveraged to provide context-aware information to field operators. To identify these variables and represent the context information generated by these variables, we develop a contextualization ontology (prefix:*ctx*), illustrated in Fig. 2 (the namespace prefixes used in this study are listed in Table B.1 in the appendix). As discussed in Section 2.1, this work divides the context into three main categories: External Context, User Context, and Interface Context. External Context encompasses data originating from the physical factory environment. Especially the data of observations and their properties traced by digital systems. We integrate *sosa:Observation* from SOSA ontology, *qudt:Quantity* from QUDT ontology, and *bot:Space* from BOT as subclasses of *ctx:ExternalContext*. The *sosa:Observation* class represents the continuous procedure of observing entities of interest. Examples include monitoring the status of machines (*isie:Machine*), tracking the properties of material batches (*isimatl:MaterialBatch*), or observing the location of personnel (*isie:Agent*). Notably, *isie:Machine*, *isimatl:MaterialBatch*, and *isie:Agent* are all reused from the InPro ontology. The *qudt:Quantity* class describes the measurement of an observable property of a particular object, event, or physical system. For instance, it can describe the load capability of the machine and the specification of the shipment (e.g., height, length, and width). The *bot:Space* class proves valuable in representing core topological concepts within environments, such as specific working zones for the field operator in factory buildings. User Context captures attributes of CAS users, such as profiles and preferences. This information is typically acquired from user input or static data sources. Interface Context pertains to the ongoing operational tasks or activities of users. In the context of the factory environment, a *ctx:User* (subclass of *isie:Agent*) executes a specific *ctx:Action* related to a particular *isipp:OperationalTask*. The *ctx:Action* represents the execution of the operational task, serving as a subclass of *ctx:InterfaceContext*. Each of these context categories serves as a subclass of *ctx:ContextualPrecondition*. This class represents the necessary prerequisites for delivering relevant context-aware information to field operators. By understanding the user, their tasks, and the surrounding environment, CAS framework can deliver targeted information that supports informed decision-making.

The contextualization ontology is further expanded within the Context-aware Platform to represent the decision-making process informed by contextual information. This extension allows the CAS framework to dynamically generate contextual information for the end-user based on the specific situation. The *ctx:Constraint* class represents the limitation governing specific operational tasks. The *ctx:Decision* class describes the process of making informed choices based on user actions (*ctx:Action*), relevant constraints (*ctx:Constraint*), and real-time contextual information from the *ctx:ExternalContext*. By considering these factors, the decision process dynamically adjusts or modifies results to be suitable for the specific context. Based on the outcome of the *ctx:Decision*, the system generates *ctx:ContextAwareInformation* tailored to the user of the CAS. This information can be delivered to the user leveraging the customized user interface, potentially including visual or audio notifications, or using the user's preferred language. Table 1 outlines the core concepts of the proposed ontological model.

Mapping the ontology to a high-level abstraction framework ensures that the terminologies employed in the model are both comprehensible to end users and enhance semantic interoperability [39]. As the top-level ontology, Basic Formal Ontology (BFO) [47] is a domain-neutral ontology designed for broad application in scientific and other fields, and it has been standardized as ISO/IEC 21838-2 [48]. Consequently, the proposed ontological model adopts BFO as its upper-level abstraction. In this model, *ctx:ContextualPrecondition* class is defined as the subclass of *obo:obo:Generically DependentContinuant* within BFO. Furthermore, the InPro ontology, also used in this study, leverages BFO as its top-level ontology, ensuring alignment and semantic consistency across these models.

4. System implementation

The platform's architecture comprises several interconnected components designed to acquire, process, reason over, and deliver contextual information to end-users (the component diagram of the proposed Context-aware Platform is depicted in Fig. C.1 in the appendix). As illustrated in Fig. 3, the components collaboratively construct a contextual information processing pipeline tailored to operational tasks.

As outlined in Section 3.3, contextual preconditions are categorized into external, user, and interface contexts, managed respectively by

Table 1
General concepts for describing the contextualization ontology.

Concepts	Definition
ContextualPreconditions	Conditions or requirements that are dependent on the specific context in which they are applied.
ExternalContext	Information derived from the physical environment.
UserContext	Attributes associated with the user.
InterfaceContext	Activities that the users are performing.
User	The user of the CAS.
Profile	User's biological and sociological information, such as height or qualifications.
Preference	User-selected system settings or configurations for customization.
Action	User's execution of a task related to a planned activity.
Constraint	A limitation that affects the way something can be done or the range of options.
Decision	The process of concluding considering various constraints.
ContextAwareInformation	Information dynamically tailored to user context or situation.

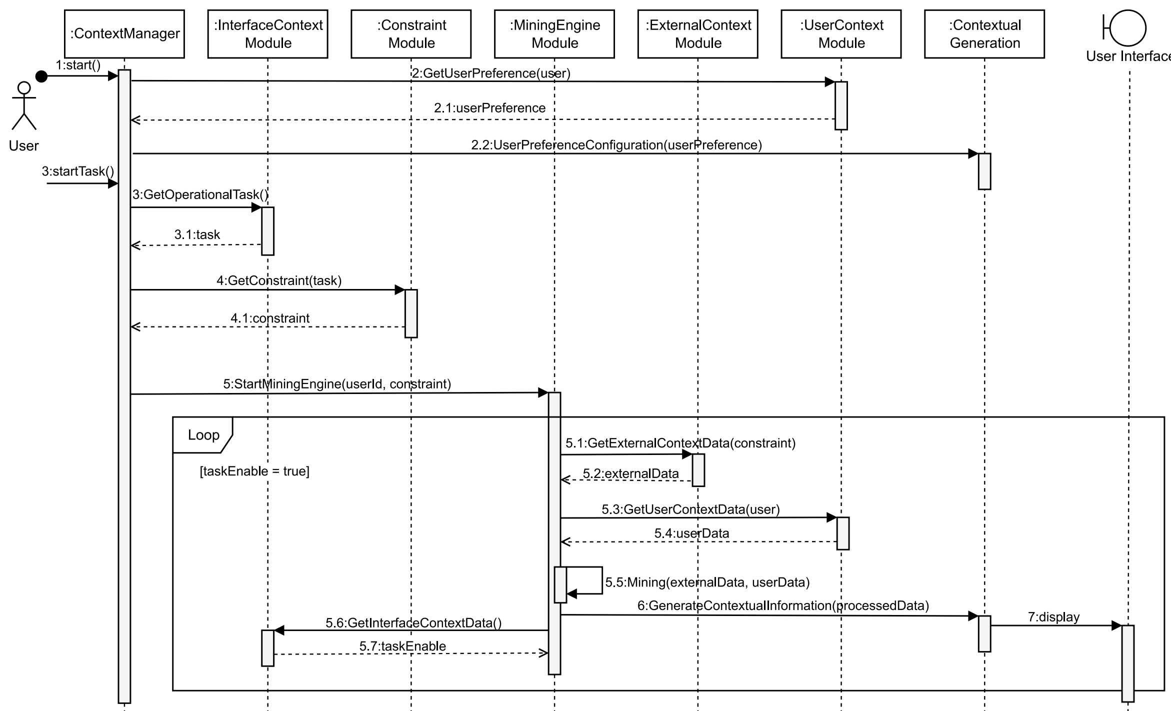

Fig. 3. Sequence diagram illustrating the usage of the proposed context-aware system.

the *ExternalContextualModule*, *UserContextModule*, and *InterfaceModule*. The *ExternalContextualModule* is tasked with retrieving contextual information based on specific constraints from graph databases within the ABox. The unified and formalized knowledge model in the TBox ensures the consistency and conformity of the data format within this module. The *UserContextModule* handles data related to user preferences and profiles. These static data sources are stored in relational databases. The user preferences parameters enable the customization and personalization of the user interface on end-user devices. The *InterfaceModule* is designed to interact with the end-user devices, monitoring the status of ongoing operational tasks. The *ConstraintModule* is notified of which constraint should be collected for the mining process related to executed operational tasks. Acting as a middleware, the *MiningEngine* processes the contextual information, performs reasoning, and delivers the refined data to the contextual generation module. This processed information is then displayed on the end devices, ensuring that the end users receive timely and relevant contextual data.

To manage the complete pipeline, the *ContextManager* is created to orchestrate each class to construct the contextual information management pipeline, as detailed in Fig. 3. Initially, the user inputs the Identification (ID) to start the system. Upon system initiation, the *ContextManager* calls the *GetUserPreference* method to fetch user preference from the *UserContextModule* and configures the user interface through the *UserPreferenceConfiguration* method. While the user starts an operational task, the *ContextManager* calls the *GetOperationalTask* method to retrieve the ongoing task information through the *InterfaceModule* and uses it as parameters to call the *GetConstraint* method to get related constraints from the *ConstraintModule*. For example, as illustrated in Fig. 5, the type name of the transportation task can be used to retrieve the corresponding constraints such as *OverLoadingConstraint*, *AccessControlConstraint*, *SafetyDistanceConstraint*, and *WithinRangeConstraint*. Subsequently, *ContextManager* initiates the *StartMiningEngine* method with the retrieved parameters (*userId* and *constraint*). During task execution, the *MiningEngineModule* runs a loop within the *StartMiningEngine* method to continuously capture information from both the *ExternalContextModule* and the *UserContextModule* for contextual information mining, while also calling the *GetInterfaceContextData* method to get task status (*taskEnable*) as the loop's condition. The processed contextual information is then delivered to the *ContextualGeneration* class for display on the end-user device. The seamless coordination among these modules facilitated by the *ContextManager* ensures a robust and efficient system for managing and utilizing contextual information.

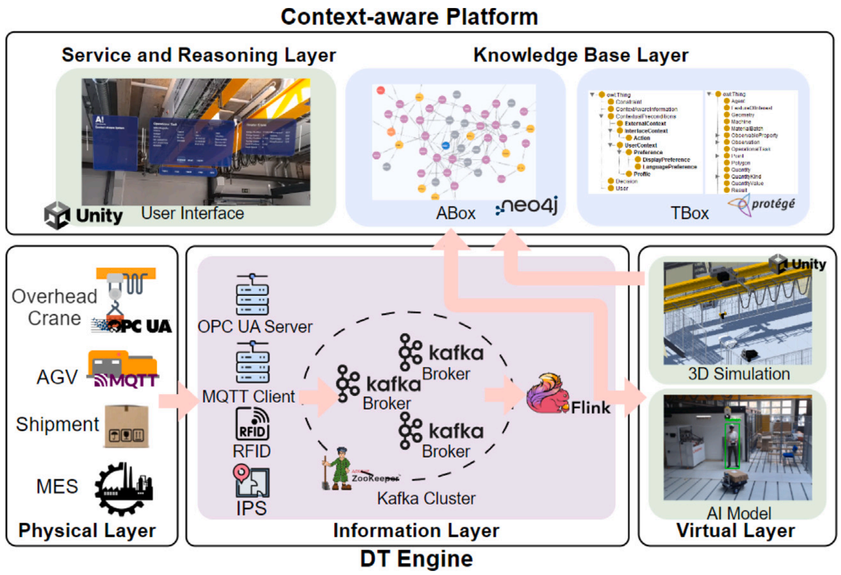

Fig. 4. Different components and technologies used for the proof-of-concept.

Table 2
Guidelines for the crane operation scenarios.

Guideline 1:
The crane shall not be loaded in excess of its rated load.
Guideline 2:
Personnel are not allowed to access the crane operation space while the crane operator is in operation.
Guideline 3:
The distance between the crane's hook and AGV should exceed a predefined safety threshold.

5. Case study: DT-driven CAS for overhead crane operation scenarios

This section presents the implementation of a DT-driven CAS to support overhead crane operators in intra-plant logistics. The system captures, processes, and delivers accurate and timely contextual information to the operator during material transfer operations. To demonstrate its capabilities, we consider a logistics scenario involving AGVs and an overhead crane.

5.1. Scenario description

Human-centered operations [49], such as intra-plant logistics, often require human execution and oversight for tasks like quality control, troubleshooting, and handling exceptions that arise during the operation process. Within manufacturing facilities, efficient logistics workflows integrate human oversight with automated machinery like AGVs and overhead cranes for seamless intra-plant material transportation. AGVs are designed for horizontal movement within a designated area, enabling human-free transportation of goods across the factory floor. Conversely, overhead cranes excel at vertically lifting and maneuvering heavy loads to reach areas with limited accessibility. These cranes are typically operated by a human onsite who is ultimately responsible for crew and crane safety. Research indicates that human error contributes to a significant portion of incidents [50]. Additionally, fatigue and human performance degradation can lead to failures in up to 80% of identified problems [51]. Although the integration of AGVs and overhead cranes enhances the efficiency of material handling, it also presents novel safety concerns for crane operators as a result of the heightened intricacy and dynamic interactions within the operational environment. This work identifies three typical safety concerns and operations in the literature regarding crane operation. These include:

- Overloading: Exceeding the crane's maximum load capacity, which can result in equipment failure [52];
- Unauthorized Access: The presence of personnel in areas where overhead cranes are operating should be strictly limited to cases of absolute necessity [53];
- Limited visibility: The risk of collisions between AGVs and overhead cranes, primarily due to a shared workspace and restricted visibility [54,55].

Building on the identified safety challenges, we have standardized them into a set of three guidelines (detailed in Table 2) within RBox to implement safety measures and ensure compliance within the operational environment. These guidelines translate the concerns regarding overloading, unauthorized access, and visibility limitations into actionable directives for the CAS. Moreover, the establishment of explicit guidelines creates accountability and helps ensure that safety protocols are implemented consistently and effectively.

5.2. Scenario implementation

This study implemented the DT-driven CAS within the Aalto Industrial Internet Campus (AIIC) to validate its effectiveness in a real-world setting. The AIIC [56], a laboratory-scale manufacturing facility equipped with an industrial overhead crane, AGVs, and supporting infrastructure, serves as the testbed. Fig. 4 provides an overview of the various components and technologies used in this scenario implementation. The DT Engine's Physical Layer integrates a range of entities: an overhead crane with an OPC UA server; AGVs with an MQTT interface; RFID-tagged shipments; a video monitor; and an open-source MES for managing the execution data.

The Information Layer within the DT Engine is required to handle streaming data. Streaming data, characterized by high velocity and variability, originates from a diverse range of sources including sensors, IoT devices, and operational systems. This data is essential for providing up-to-date insights into the physical system's dynamic state, such as current operating conditions, equipment status, and ongoing processes. Apache Kafka and Apache Flink were employed in this research to efficiently process high-volume, real-time data streams. Kafka's distributed architecture, coupled with Kafka Connect, facilitated seamless data ingestion from diverse sources. Flink's low-latency processing capabilities enabled timely data analysis. To ensure scalability across large-scale factory environments with an increasing number of data sources, Kafka's ability to add brokers and partitions was

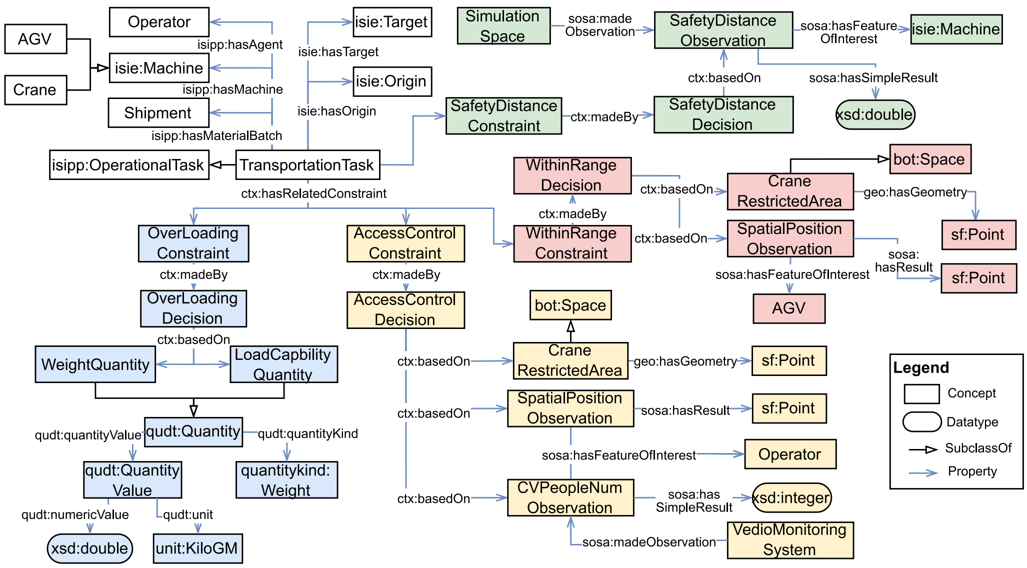

Fig. 5. The classes and relationships pertaining to the intra-plant material transfer use case. The rectangular box with blue color represents the concepts and relationships for the *Overloading* Decision, the yellow box represents the *AccessControl* Decision, the red box represents the *WithinRange* Decision, the green box represents the *SaftyDistance* Decision.

leveraged [57]. Similarly, Flink's distributed data processing ensures that performance remains consistent even as data volumes grow significantly. As illustrated in Fig. 4, data collection involved a variety of protocols and technologies. The OPC UA server was established to communicate with the overhead crane, MQTT to interface with AGVs, and RFID to track shipment status. A WebSocket-based connection facilitated data capture from the Indoor Positioning System (IPS). To manage the flow of data, a Kafka cluster with three brokers was deployed, orchestrated by Apache ZooKeeper. Flink applications were then responsible for processing the incoming data streams based on the defined data format in the TBox and integrating the processed data into a Neo4j graph database. The DT Engine's Virtual Layer, elaborated upon in subsequent use cases, incorporates a Unity-based virtual simulation environment and a YOLO (a real-time object detection system [58])-powered video monitoring system to provide additional insights into the system.

For the Context-aware Platform, the knowledge models were constructed using OWL within the Protégé [59]. Fig. 5 illustrates the classes and relationships of the knowledge models for crane operation use cases. Entities like Operator, Crane, AGV, and Shipment inherit from InPro classes (*isie:Agent*, *isie:Machine*, *isimatl:MaterialBatch*), whereas *TransportationTask* (a subclass of *isipp:OperationalTask*) interconnects these entities. Within a *TransportationTask*, the classes *isie:Target* and *isie:Origin* define the source and destination points for the transportation process. In the context of overhead crane operations, several relevant constraints were established: *OverLoadingConstraint*, *UnauthorizedMonitoringConstraint*, *SafetyDistanceConstraint*, and *WithinRangeConstraint* to meet three guidelines for safe operation. For instance, *OverLoadingConstraint* corresponds to guideline 1, *AccessControlConstraint* aligns with guideline 2, and both *WithinRangeConstraint* and *SafetyDistanceConstraint* contribute to guideline 3. These constraints leverage external contextual information derived from quantities (*qudt:Quantity*), observations (*sosa:Observation*), and building topology information (*bot:Space*) to inform decision-making processes represented by *ctx:Decision*. The following section details use cases demonstrating the application of these constraints and their corresponding decisions. The QUDT ontology provides a foundation for representing quantities and their corresponding units. Within this case study, weight-related dimensions are modeled using subclasses of *qudt:Quantity*. Specifically, *LoadCapabilityQuantity* (associated with the *Crane* class) and *WeightQuantity* (associated with the *Shipment* class) describe the crane's load capacity and the weight of a shipment, respectively. They both utilize the kilogram unit (*unit:KiloGM*) and a double format (*xsd:double*) for the value. The *CraneRestrictedArea* class inherits from *bot:Space* and is further represented using *sf:Point* (a subclass of *geo:Geometry*) to provide a rich geospatial representation. The *sosa:Observation* class signifies the act of measuring or estimating a property's value. Leveraging ICT solutions, the subclass *SpatialPositionObservation* enables representing the spatial positions of mobile entities (e.g., AGVs and operators). The *sosa:Observation* class extends beyond representing physical entities. It also allows for the representation of data observation from systems within the DT Engine. For example, *SafetyDistanceObservation* (a subclass of *sosa:Observation*) describes distances between machines measured through real-time 3D simulations. Additionally, *CVPeopleNumObservation* represents the number of people detected by an AI-powered video monitoring system within a restricted zone. This contextualization ontology, built upon existing ontologies, ensures clear and standardized data formats. This, in turn, enables seamless communication between different variables at the semantic level. This semantic interoperability is crucial for effective contextual information management and reasoning within the platform.

The TBox, implemented in OWL, defines the ontology's schema, including classes, properties, and relationships that establish the domain's conceptual foundation. OWL's expressiveness and reasoning capabilities facilitate the creation of a detailed and semantically consistent model, enabling advanced inferencing and consistency checking. However, for managing the ABox, which houses the actual data instances conforming to the TBox schema, Neo4j offers distinct advantages. Neo4j excels at handling large volumes of interconnected data with exceptional query performance and scalability [60]. By storing the ABox in Neo4j, the platform leverages its efficient graph traversal algorithms and Cypher query language to achieve rapid retrieval and analysis of instance data. For the user interface, AR has emerged as an effective tool for enhancing the user experience of manufacturing field operators [61]. By overlaying digital information onto the physical environment, AR provides operators with real-time guidance, interactive manuals, and step-by-step instructions, significantly streamlining complex assembly processes. Furthermore, the hands-free nature of AR devices enables operators to access critical contextual information without disrupting their workflow. To leverage these advantages, the Service and Reasoning Layers were implemented using the Unity

Fig. 6. Interface of the proposed CAS.

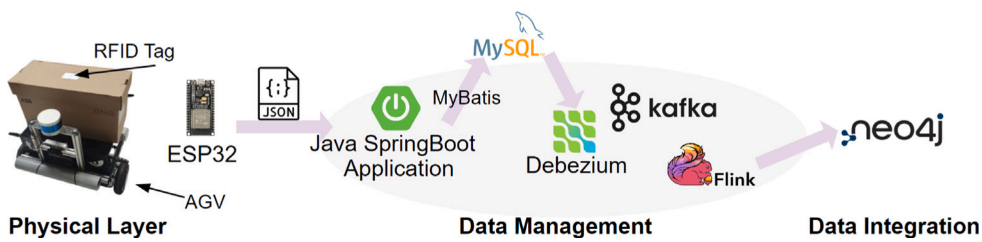

Fig. 7. The workflow of material information integration.

Engine, adhering to the system architecture outlined in Section 4. This study utilized Microsoft HoloLens 2 as the hardware platform, and a user interface was developed within the Unity environment, as illustrated in Fig. 6. The user interface employs a tri-panel design to enhance operator efficiency. The left panel visualizes contextual information generated to assist the crane operator. The center panel integrates a task management application directly within the AR interface. This application allows the operator to retrieve task details and update their status within the MES database. Operators can leverage their hands to interact with a slider control at the bottom of the center panel to check the details of their assigned tasks. Additionally, they can utilize on-screen buttons to efficiently record the time duration for ongoing tasks and update task progress in real-time. The right panel functions as a dashboard, providing real-time visualization of the crane's operational status.

5.3. Use case 1: Overloading checking

A challenge in overhead crane operations is exceeding the crane's lifting capacity (overloading). This often occurs due to inaccessible weight data. Shipment weight and crane capacity reside in separate systems with incompatible data formats. This fragmented data prevents crane operators from making informed decisions. Fig. 7 depicts the material information integration workflow. Within the factory floor, upon delivery by an AGV on the factory floor, an RFID reader scans the attached RFID tag on the parcel. An ESP32 microcontroller then transmits the extracted information, formatted in JSON, to a Java SpringBoot application. This application receives the shipment data and stores it within a MySQL database using MyBatis services. Data synchronization is achieved through Debezium, an open-source platform for change data capture. Any modifications within the MySQL database are captured by Debezium and forwarded to the Apache Kafka data pipeline. Subsequently, Flink processes the data based on the knowledge model defined in the TBox. Finally, the processed material information and its relationships are updated in the Neo4j.

The *OverLoadingConstraint* class is defined for this use case to meet guideline 1. *OverLoadingDecision* made by the *OverLoadConstraint* is based on the *LoadCapabilityQuantity* of the *Crane* and the *WeightQuantity* of the *Shipment*. The Information Layer processes the data (shipment weight and crane load capacity) aligning a unified data format. This data is represented by *qudt:Quantity* with the weight dimension *quantitykind:Weight*. Query 1 in Table D.1 in the appendix illustrates the query for overloading checking. By leveraging this logic, the *MiningEngineModule* can automatically verify if a shipment's weight surpasses the crane's hoisting capability. In the event of an overloading situation, CAS will visualize notification information to the operator.

5.4. Use case 2: Access control

To ensure the safety of personnel during crane operation, the CAS employs a multi-layered approach for access control within the designated operational area. The system leverages an IPS and an AI-powered video monitoring system. The IPS utilizes Ultra-Wide Band (UWB) radio technology in conjunction with a cloud-based positioning engine for precise localization. Authorized personnel wear detectable tags that transmit their spatial location within the operational area via physical beacons to this positioning engine for calculation. A WebSocket service is employed to retrieve these spatial coordinates through the WebSocket API. Subsequently, the Information Layer translates the coordinates into a standardized format using a Coordinate Reference System (CRS). Subsequently, it updates the values of instances belonging to the class *SpatialPositionObservation*.

While granting tag access to all production personnel (e.g., suppliers, maintenance workers, inspectors) simplifies access management, it introduces a potential security vulnerability. To address this vulnerability, CAS further utilizes an AI-powered video monitoring system for real-time human detection. The AI model in Fig. 4 is the screenshot of AI-enhanced human detection. The monitoring system updates the value to the instance of the *CVPeopleNumObservation* class, reflecting the number of people detected within the predefined operational area. The Mining Engine Module of the Context-aware Platform analyzes the

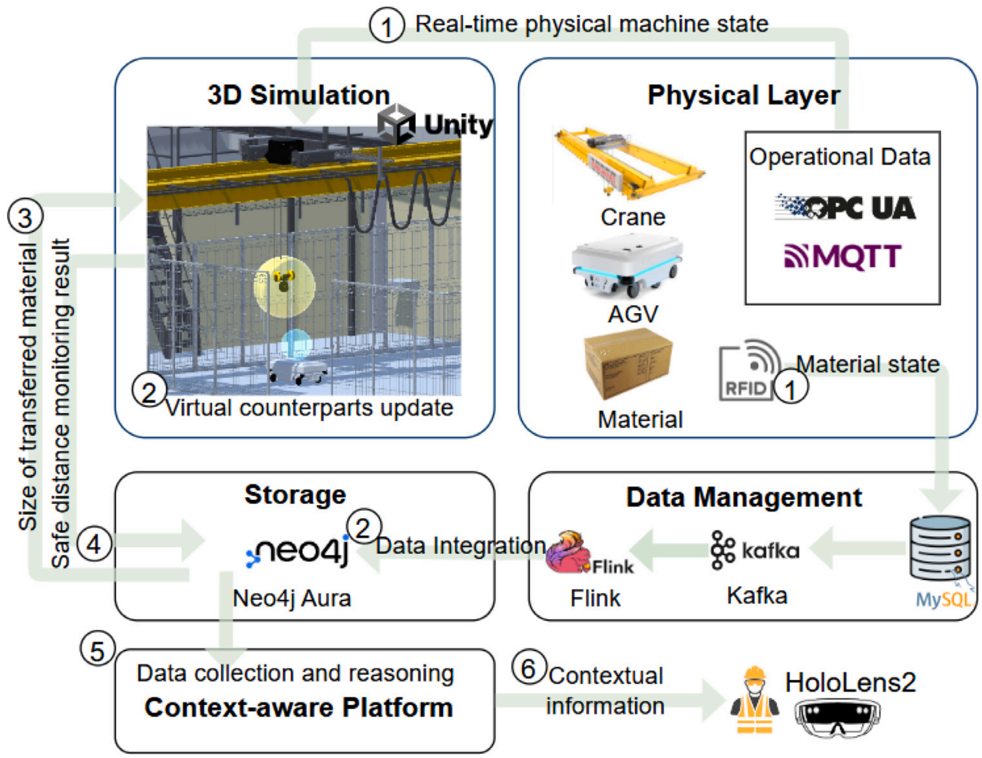

Fig. 8. The workflow of Use Case 3 for collision detection. Step 1: capture data from the physical environment; Step 2: update virtual counterparts and data integration in Neo4j; Step 3: retrieval specifications of transferred material and update the size of the safety zones; Step 4: update safe distance monitoring result to Neo4j; Step 5: data collection and reasoning; Step 6: contextual information delivery.

perceived data from both systems, including the spatial positions of authorized personnel (from IPS) and the total number of people detected by the video system. The *AccessControlConstraint* class is defined to meet guideline 2. The *AccessControlDecision* made by the *AccessControlConstraint* is based on three external contexts, including the *CVPeopleNumObservation* of the video system, the *Location* of the crane working space, and the *SpatialPositionObservation* of the operators. The Cypher spatial function (*point.withinBBox*) can be employed to determine if an operator's CRS-based position (obtained from the UWB IPS) falls within a predefined zone. This function further facilitates counting the total number of detected personnel within the operational area. The CAS triggers alerts for the crane operator under two scenarios: (1) when the number of people detected by the camera exceeds one, or (2) when the number of people detected by the IPS exceeds one, as shown in Table D.1 in the appendix (Query 2). By effectively utilizing existing infrastructure (cameras and IPS) and incorporating AI for real-time analysis with Cypher spatial functions, CAS offers a robust and efficient solution for access control, enhancing safety during crane operations.

5.5. Use case 3: Visibility enhancement

Traditional crane operation methods rely on human operators to identify potential collisions, but increasingly complex work environments with AGVs pose challenges due to limited visibility and blind spots. Crane operators may not always be aware of AGV activity, leading to accidental interactions. In shared workspaces, constant awareness is required to prevent collisions between AGVs and overhead cranes. Our CAS leverages an IPS and a simulation environment to enhance operator awareness of surrounding AGVs. Similar to the previous use case, the IPS captures the coordinates of AGVs, providing CAS with their real-time spatial positions. As shown in Query 3 of Table D.1 in the appendix, the *DebotWithinIlmatarWSConstraint* makes a *WithinRangeDecision* based on the location of the crane's working space and the AGV coordinates. If an AGV enters the crane's working space, the *MiningEngineModule* triggers a notification for the operator.

However, unlike AGVs with primarily horizontal movement with 2D coordinate systems, the crane's hook exhibits 3D motion capabilities, vertically and horizontally. This disparity in coordinate systems between the crane's hook and AGVs presents a significant challenge for calculating real-time clearance, hindering effective collision prevention. Our CAS leverages a real-time virtual environment within the DT Engine. Fig. 8 illustrates the workflow designed for collision detection in dynamic manufacturing environments. The Physical Layer comprises devices such as cranes, AGVs, and materials, which generate data through OPC UA, MQTT, and RFID systems. Building on our prior research [56], real-time operational data from physical cranes and AGVs are continuously transmitted to MQTT and OPC UA clients within Unity. This data stream ensures that the game engine can update the state of the virtual counterparts of physical machines. The update rate in Unity is set to 0.02 s per frame, ensuring a highly responsive system. Structured data related to materials is ingested and managed in a MySQL database, with real-time data flow facilitated by Kafka for streaming and Flink for low-latency processing. Neo4j Aura is utilized to store and query complex relationships between entities. The 3D Simulation Layer, powered by Unity, mirrors the physical system and integrates the Building Information Model (BIM) from AIIC, 3D models of AGVs, and a crane model in FBX format. This virtual environment continuously performs calculations and updates the distance values for instances of the *SafetyDistanceObservation* class, which are stored in Neo4j Aura. The External Context Module of the CAS uses the *SafetyDistanceObservation* class to query these distance values, as shown in Table D.1 in the appendix (Query 4). The Mining Engine Module within the Context-aware Platform then compares these distance values to predefined safety thresholds and sends results to Contextual Generation. Finally, HoloLens2 provides operators with AR visualizations,

Table 3
The responsiveness time of scenarios.

Scenario	Mean	Range	SD
(1) MySql to Neo4j (ms)	270.20	501.48	0.15
(2) MQTT to Neo4j (ms)	47.11	35.57	0.01
(3) OPC UA to Neo4j (ms)	49.37	37.80	0.01
(4) Neo4j to Context-aware Platform (ms)	238.29	955.48	0.23

enabling user-centered interactions with the system to enhance safety and operational efficiency.

6. Effectiveness analysis

To comprehensively evaluate the effectiveness of the proposed CAS, we designed three tests: system responsiveness, task-based information retrieval, and usability testing, each addressing key challenges in dynamic manufacturing environments. These tests are essential for measuring the system's ability to handle rapidly changing conditions, integrate heterogeneous data sources while maintaining data consistency, and deliver personalized, real-time information to users. Together, these evaluations provide a holistic assessment of the CAS, ensuring it meets the demands of operational efficiency in complex, dynamic settings.

6.1. Evaluation of system responsiveness

To assess the responsiveness of our CAS, we measured the time it takes for data changes to be processed and made available for use. This evaluation focused on two key latency components: data source to Neo4j and Neo4j to Service Layer. The former quantifies the delay between data modifications in various sources (MySQL, MQTT, and OPC UA) and their incorporation into the Neo4j graph database. The latter measures the time elapsed between a Neo4j data update and its arrival at the *GenerateContextualInformation* method within the CAS, which is responsible for processing and generating contextual information for visualization. Four scenarios were defined to evaluate system responsiveness: Scenario 1 evaluates the end-to-end latency from data reception in the MySQL database to its integration into Neo4j, Scenario 2 measures the latency from a data change in the MQTT client to its integration into Neo4j, Scenario 3 assesses the latency from a data change in the OPC UA server to its integration into Neo4j, and Scenario 4 evaluates the latency from a data change within Neo4j to the *GenerateContextualInformation* method in the Context-aware Platform. This method is responsible for processing and generating contextual information for visualization purposes. A desktop computer equipped with an Intel Core i9 processor and 64 GB RAM hosted a three-broker Kafka cluster for real-time data stream processing. A standalone Neo4j database instance operated on a separate laptop within the local network. For each scenario, 100 data points were collected to calculate average latency. Table 3 presents the latency values observed for each scenario, including the mean, Standard Deviation (SD), and range. The results indicate an average latency of 270.20 $milliseconds$ (ms) for Scenario 1, which involves data transfer from the MySQL database. The latency for Scenario 2 (MQTT client) is significantly lower at 47.11 ms, and Scenario 3 (OPC UA server) exhibits a similar latency of 49.37 ms. Finally, the latency from Neo4j to the information visualization process (Scenario 4) is 238.29 ms. Considering the average human reaction time of approximately 500 ms [62], our target system latency is set at 1 $second$ to allow sufficient time for human operators to react to critical events. The measured total latency for data propagation through the system (ranging from 285.40 ms to 508.49 ms) falls within acceptable bounds.

6.2. Task-based information retrieval

In this section, we evaluate the utility of the proposed system for managing heterogeneous data sources to ensure data consistency and quality. To assess the effectiveness of information integration, a task-based approach was implemented using multiple questions presented in Table 4. These questions are specifically designed to test the CAS's ability to retrieve accurate and relevant information for a crane delivery task, thereby validating its performance in real-world applications.

Table E.1 in the appendix presents five Cypher queries corresponding to the questions outlined in Table 4. The first query addresses Question 1 by retrieving the crane's ongoing operation task. Query 2 responds to Question 2, identifying the height of the shipment delivered by the crane, while Query 3, aligned with Question 3, retrieves the human operator's states of this operational task. These first three queries verify the system's ability to interconnect multiple entities, such as tasks, machines, and materials. Query 4 answers Question 4 by retrieving the distance between the crane and the AGV, calculated by the 3D simulation environment. Finally, Query 5 addresses Question 5 by checking the number of humans within the AIIC crane's working environment, as detected by the AI video monitoring system. These last two queries validate the system's ability to retrieve information from diverse data sources. This evaluation confirms the system's effectiveness in delivering contextually relevant information in dynamic, real-world environments, demonstrating its capability to handle complex, interconnected data.

6.3. Usability test

A usability evaluation was conducted to assess the effectiveness of the proposed CAS for overhead crane operations. 10 participants from Aalto University (8 male, 2 female; age range 22–34 years) participated in the test. They have been trained and certificated to operate the overhead crane. Participants were asked to complete two material delivery tasks operating an overhead crane. They wore a HoloLens 2 Trimble XR10 that is running our CAS. During each operation, personnel and AGVs entered the work area randomly, with CAS providing contextual information to the participants. System usability was evaluated using the System Usability Scale (SUS) [63]. The SUS is a well-established, 10-item questionnaire with a 5-point Likert scale that measures overall system usability (strongly agree to strongly disagree).

The SUS test yielded a mean usability score of 78, indicating a positive reception by users. According to Bangor et al.'s adjective ratings, a score of 78 falls between "Good" and "Excellent", placing the product in the 80th to 85th percentile for usability. While the standard deviation of 10.261 suggests a relatively consistent user experience, the range of 27.5 points highlights variability in user perceptions. The result of the usability test is illustrated in Table F.1 in the appendix. Overall, the results demonstrate a solid usability foundation for our proposed CAS.

7. Discussion, limitations, and future works

The proliferation of Information and Communication Technologies (ICT) in modern manufacturing facilities has generated an abundance of real-time data, which is crucial for optimizing operations. The pivotal role of context-aware applications in analyzing and interpreting this data has been previously emphasized [3]. However, there remains a significant gap in providing field operators with solutions that adequately capture and visualize the manufacturing context in a dynamic and evolving environment. Existing solutions for manufacturing context representation often fall short in terms of adaptability and interoperability. For instance, while some works have introduced operational context models [15] and explored autonomous systems based on DT [16], these models frequently lack the comprehensive

Table 4
Specified questions and answers for the information retrieval test.

Specified Competency questions	Answers
(1) What's the ongoing operational task of the overhead crane 'Ilmatar'?	"Transport32"
(2) What's the shipment size for the overhead crane 'Ilmatar'?	"Shipment32", 47.00, "Centimeter"
(3) What's the human operator's ID and spatial position for the operational task 'Transport32'?	"001", "point(srid:7203, x:10, y:23)"
(4) What's the distance between the overhead crane 'Ilmatar' and the AGV 'Debot'?	"AIIC_Virtual_System", 534.35
(5) How many people are within the AIIC crane working environment?	1

approach necessary to support the complexity of modern manufacturing environments. Similarly, surveys on CAS have highlighted the need for a more holistic framework that captures the dynamic nature of industrial settings [17]. Our proposed CAS addresses these limitations by integrating DT and AR technologies with an ontology-driven approach to contextualization, creating a solution that is more adaptable, interoperable, and user-centric than existing approaches. This novel CAS enhances decision-making within dynamic production environments, bridging the gap in current research and offering a more effective tool for field operators.

By leveraging data from DTs, the proposed CAS gains a holistic understanding of the manufacturing ecosystem. This comprehensive view fosters seamless communication between the platform and diverse industrial systems, regardless of their origin or implementation details. The DT Engine's inherent adaptability allows the CAS to accommodate changes within the manufacturing environment without requiring significant reconfiguration. This flexibility is crucial for navigating the dynamic nature of production facilities, where processes and layouts can evolve rapidly. Real-time updates from DTs empower the CAS to adjust its decision-making logic, accounting for these dynamic changes in the environment. Furthermore, the virtual layer of the DT Engine enables advanced analytics such as calculating precise distances between machines within the 3D simulation environment and employing AI-driven human recognition through video analysis. The integration of DT data into the CAS significantly enhances the system's ability to provide accurate, timely, and relevant contextual information to support human operators in making informed decisions.

Our research presented an ontology model for contextualization within the manufacturing domain. This model aims to systematically represent surrounding information for field operators, facilitating interoperability and scalability crucial for dynamic manufacturing environments. Although existing models provide some solutions [15–17], they often lack the adaptability needed for the system. Our ontology model addresses these gaps by categorizing context into three main aspects: external context, user context, and interface context. The external context category is considered the most critical aspect, as it provides a foundation for representing real-world entities and phenomena relevant to manufacturing operations. These classes serve as formalized concepts to describe the diverse variables encountered within the manufacturing domain. By integrating these aspects, we achieve a more holistic and adaptable representation of contextual information, improving semantic interoperability and, as a result, supporting field operators in dynamic contexts. The proposed contextualization ontology is further extended to encompass the decision-making process within the CAS. This extension provides guidelines for the software architecture, facilitating a comprehensive approach to managing contextual information within decision-making workflows.

This research introduced a user interface leveraging AR technology to visualize contextual information for field operators in a hands-free and intuitive manner. The AR interface minimizes disruption to the operator's workflow by presenting relevant contextual data overlaid on the real world through an AR headset. To further enhance user experience, the system architecture incorporates the *UserContext* module within the Context-aware Platform. This module considers user preferences by querying and integrating them into the AR interface configuration. This personalization ensures that each user receives the most relevant information tailored to their needs and working style. Additionally, the AR interface seamlessly integrates task management and machine operational data, providing operators with a comprehensive situational overview (see Fig. 6). Usability testing yielded a score of 78, affirming the effectiveness of the proposed AR interface. These results validate the interface's feasibility and its potential to enhance operational efficiency and decision-making for human operators in dynamic manufacturing settings.

The practical implications of our approach extend beyond academic contributions. In industry, the integration of DT technologies into CAS is expected to revolutionize decision-making processes by providing real-time insights into the manufacturing ecosystem. This system enables field operators to better understand ongoing processes, reducing downtime, enhancing productivity, and improving operational efficiency. The flexibility of our solution, which accommodates rapidly evolving production environments, is key to maintaining efficiency in modern factories.

This research has also identified the following limitations. First, the effectiveness of the proposed CAS depends on the existing digital infrastructure of the manufacturing environment. Facilities with outdated or incompatible systems may face significant hurdles in adopting the new technology. Before implementation, assessing the existing infrastructure's capabilities is paramount. The utilization of IoT technologies is one solution to bring intelligence to legacy machines [64]. Upgrading or modifying systems may be necessary to ensure seamless integration and maximize the benefits of the proposed CAS.

Data security and privacy are paramount for any CAS collecting data in an IoT environment [19]. The CAS gathers data from industrial operators regarding their preferences and profiles. The data provided contains sensitive and proprietary information, including preferences and profiles. Implementing privacy safeguards is crucial for preventing unauthorized access and compiling regulations such as the General Data Protection Regulation (GDPR) [65] is essential for establishing trust with end users. This work prioritizes data security by leveraging existing safety mechanisms from protocols and technologies like OPC UA, MQTT, and Kafka. These protocols offer built-in features such as encryption, authentication, and access control, ensuring data confidentiality and integrity. OPC UA, for example, utilizes encryption and authentication certificates for secure industrial data exchange. MQTT employs Transport Layer Security (TLS) encryption and username/password authentication for secure communication. Similarly, Kafka provides Access Control Lists (ACLs) and Secure Sockets Layer (SSL)/TLS encryption for data transmission security. While these measures bolster the platform's security, a more holistic approach is necessary. Future iterations will focus on a robust security framework to achieve comprehensive protection against evolving cyber threats. This includes implementing end-to-end encryption across all communication channels and employing a unified identity and access management system. A holistic security architecture will safeguard the entire CAS, fostering a trustworthy environment for data collection, processing, and utilization.

Furthermore, this study demonstrated the ability to utilize rule-based reasoning to extract relevant contextual insights in dynamic manufacturing environments. Although demonstrating flexibility through several use cases is a crucial initial step, the current approach relies on manually extracting constraints from regulations. This method becomes impractical for complex and standardized industrial operations, which necessitate a scalable and efficient mechanism for rule management.

This includes establishing a method for efficiently regulating rules, translating regulations and standards into executable rules within the platform, and implementing robust storage and update mechanisms for the rule base. Addressing these complexities will ensure the reasoning engine's adaptability and effectiveness in the face of evolving manufacturing scenarios.

Our proposed system demonstrates advancements in interoperability and adaptability for CAS within manufacturing environments. While extensive pressure testing is not conducted in this work, real-world manufacturing scenarios involve high-volume data streams. To ensure consistent performance under such conditions, the system's scalability and the DT Engine's efficiency require further evaluation. This includes investigating dynamic computation resource allocation strategies that can adapt to varying data loads within large and complex manufacturing environments. Addressing these scalability considerations will be critical for ensuring the platform's effectiveness in real-world industrial applications.

Future research will focus on enhancing the platform's autonomy, usability, applicability, and standardization across a broader spectrum of domains. A pivotal objective is to establish a bidirectional communication channel with the DT Engine, transforming the CAS from a passive visualization tool to an active decision-maker capable of proactively responding to dynamic contextual changes. This evolution is anticipated to significantly enhance the platform's efficacy and responsiveness within complex manufacturing settings. To optimize user experience and interaction, the development of a user-centric interface incorporating intuitive modalities such as voice and gesture control will be prioritized. This approach aims to enhance accessibility, particularly in operational environments where traditional input methods may be impractical or hazardous. Furthermore, the integration of advanced AI techniques for data analysis and solution generation is a key focus area. The implementation of real-time anomaly detection systems, leveraging both historical and real-time data, will be explored to proactively identify and mitigate potential production disruptions. Furthermore, the application of the CAS to a broader array of human-centric manufacturing scenarios — such as maintenance, inspection, and assembly — is envisioned. As part of our standardization efforts, future work will align the DT Engine with the ISO 23247 Digital Twin Manufacturing Framework [66], ensuring compliance with widely accepted industrial standards. This alignment will promote interoperability and scalability, supporting a more cohesive and standardized approach to DT development in manufacturing environments.

8. Conclusion

To assist the manufacturing field operators in efficiently perceiving their dynamic working environment, this work proposed a DT-driven CAS. DT enables a digital representation of the complex factory environment. Leveraging AI and 3D simulation, the system enables providing additional information. Within the Context-aware Platform, an ontology-based context model is introduced to integrate diverse contextual information within the factory environment, including external context, user context, and interface context. The implemented platform is employed to offer adaptable contextual insights tailored to field operators' tasks. An AR interface provides field operators with real-time contextual visualization to assist them in their on-site operations. By combining DT, semantic technologies, and AR, the proposed CAS bridges the gap between physical and digital realms, facilitating seamless communication and collaboration across manufacturing systems. This innovative approach enhances the adaptability, interoperability, and user-centricity of CAS. Through the implementation of the practical use case, the proposed system has demonstrated its capability to perceive information from heterogeneous sources, adapt to changing surrounding states, and deliver contextual information to the end user for informed decision-making, thereby validating the conceptual framework introduced in this research. This solution addresses the identified research problem and provides a step towards enhancing efficiency and safety in manufacturing operations through the fusion of advanced technologies and human expertise.

CRediT authorship contribution statement

Chao Yang: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Hao Yu:** Writing – review & editing, Writing – original draft, Methodology, Conceptualization. **Yuan Zheng:** Writing – review & editing, Writing – original draft, Methodology, Formal analysis, Conceptualization. **Lei Feng:** Writing – review & editing, Validation, Methodology. **Riku Ala-Laurinaho:** Writing – review & editing, Supervision. **Kari Tammi:** Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Chao Yang reports financial support was provided by Business Finland.

Acknowledgments

This work was supported by the NECOVERSE project (10768/31/2022), a co-innovation joint project funded by Business Finland.

Appendix A. List of abbreviations and acronyms.

See Table A.1.

Appendix B. List of Namespace prefix

See Table B.1.

Appendix C. Component diagram of the proposed context-aware platform.

See Fig. C.1.

Appendix D. Cypher queries within the context-aware platform

See Table D.1.

Appendix E. Cypher statements for information retrieval

See Table E.1.

Appendix F. Usability test result

See Table F.1.

Data availability

Data will be made available on request.

Table A.1
List of abbreviation and acronyms used in the paper.

Abbr.	Definition	Abbr.	Definition
ABox	Assertion Box	IT	Information Technology
ACLs	Access Control Lists	MQTT	Message Queuing Telemetry Transport
AIIC	Aalto Industrial Internet Campus	NFC	Near Field Communication
AR	Augmented Reality	OPC UA	OPC Unified Architecture
BIM	Building Information Model	OT	Operational Technology
BOT	Building Topology Ontology	OWL	Web Ontology Language
CAS	Context-Aware System	QUDT	Quantities, Units, Dimensions and Types
CPS	Cyber-Physical System	RBox	Rule Box
CRS	Coordinate Reference System	RDF	Resource Description Framework
DT	Digital Twin	RFID	Radio-Frequency Identification
GDPR	General Data Protection Regulation	SD	Standard Deviation
HRC	Human–Robot Collaboration	SOSA	Sensor, Observation, Sample, and Actuator
HTTP	Hypertext Transfer Protocol	SSL	Secure Sockets Layer
ICT	Information and Communication Technologies	SUS	System Usability Scale
ID	Identification	TBox	Terminological Box
InPro	Industrial Production workflow ontology	TLS	Transport Layer Security
IoT	Internet of Things	UWB	Ultra-Wide Band
IPS	Indoor Positioning System		

Table B.1
The list of the namespace prefix.

@prefix bot: <https://w3id.org/bot#>.
@prefix ctx: <https://w3id.org/contextualization#>.
@prefix geo: <http://www.opengis.net/ont/geosparql#>.
@prefix isie: <https://w3id.org/IndustrialSystemIntegration/Entities#>.
@prefix isipp: <https://w3id.org/IndustrialSystemIntegration/ProductionProcess#>.
@prefix obo: <http://purl.obolibrary.org/obo/>.
@prefix qudt: <http://qudt.org/schema/qudt/>.
@prefix sf: <http://www.opengis.net/ont/sf#>.
@prefix sosa: <http://www.w3.org/ns/sosa/>.
@prefix unit: <http://qudt.org/vocab/unit/>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.

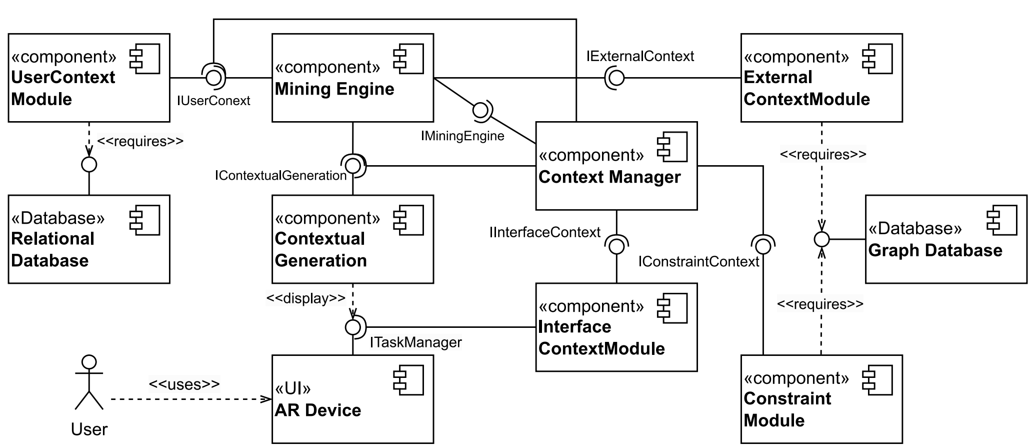

Fig. C.1. Component diagram of our Context-aware Platform.

Table D.1
Cypher queries within the Context-Aware Platform for different constraints checking.

Query 1:
```
MATCH (d:Decision)-[:madeBy]→(:Constraint {hasName:'SP1OverLoadingConstraint'}),
(d)-[:basedOn]→(w:Quantity)-[:quantityKind]→(:Weight),
(d)-[:basedOn]→(l:Quantity)-[:quantityKind]→(:LoadCapability),
(w)-[:quantityValue]→(v1:QuantityValue),
(l)-[:quantityValue]→(v2:QuantityValue)
RETURN d.hasName AS Decision,
CASE WHEN v1.value > v2.value THEN true ELSE false END AS Result
```
Query 2:
```
MATCH (d:Decision)-[:madeBy]→(:Constraint {hasName: "IlmatarWSAccessControlConstraint"}),
(d)-[:basedOn]→(l:Location),
(l)-[:hasPoint]→(p1:LowerLeftPoint),
(l)-[:hasPoint]→(p2:UpperRightPoint),
(d)-[:basedOn]→(:Observation)-[:hasResult]→(r:Result),
(d)-[:basedOn]→(:Observation)-[:hasResult]→(a:Point),
WHERE point.withinBBox(a.value, p1.value, p2.value)
WITH COUNT(*) AS IPS, r.value AS CV, d
RETURN d.hasName AS Decision,
CASE WHEN IPS > 1 OR CV > 1 THEN true ELSE false END AS Result
```
Query 3:
```
MATCH (d:Decision)-[:madeBy]→(:Constraint {hasName: 'DebotWithinIlmatarWSConstraint'}),
(d)-[:basedOn]→(l:Location),
(l)-[:hasPoint]→(p1:LowerLeftPoint),
(l)-[:hasPoint]→(p2:UpperRightPoint),
(d)-[:basedOn]→(:Observation)-[:hasResult]→(p3:Point)
RETURN d.hasName AS Decision, point.withinBBox(p3.value, p1.value, p2.value) AS Result
```
Query 4:
```
MATCH (d:Decision)- [:madeBy]→(:Constraint{hasName: "DebotIlmatarSafetyDistanceConstraint"}),
(d)-[:basedOn]→(o:Observation)-[:hasResult]→(r:Result)
RETURN d.hasName AS Decision, r.numericalValue AS Result
```

Table E.1
Cypher statements for the validation questions.

#Query 1
```
MATCH (tt:TransportationTask)-[:hasMachine]→(m:Machine)
WHERE m.hasName = 'IlmatarCrane' and tt.hasStatus = 'Ongoing'
RETURN tt.hasName
```
#Query 2
```
MATCH (tt:TransportationTask)-[:hasMachine]→(m:Machine),
(tt)-[:hasMaterialBatch]→(mb:MaterialBatch)-[:hasSpecification]→(s:Specification),
(s)-[:hasHeight]→(h:Height)←[:quantityKind]-(:Quantity)-[:quantityValue]
→(qv:QuantityValue)-[:unit]→(u:Unit)
WHERE m.hasName = 'IlmatarCrane' and tt.hasStatus = 'Ongoing'
RETURN mb.hasName, qv.value, u.hasName
```
#Query 3
```
MATCH (ot:OperationalTask)-[:hasAgent]→(a:Agent),
(spo:SpatialPositionObservation)-[:hasFeatureOfInterest]→(a),
(spo)-[:hasResult]→(p:Point)
WHERE ot.hasName = 'Transport32' RETURN a.hasId, p.value
```
#Query 4
```
MATCH (sd:SafetyDistanceObservation)-[:hasFeatureOfInterest]→
(:FeatureOfInterest{hasName:'IlmatarCrane'}),
(sd)-[:hasFeatureOfInterest]→(:FeatureOfInterest{hasName:'Debot'}),
(s:System)-[:madeObservation]→(sd),
(sd)-[:hasResult]→(r:Result) return s.hasName, r.value
```
#Query 5
```
MATCH (cv:CVPeopleNumObservation)-[:hasFeatureOfInterest] →
(fi:FeatureOfInterest{hasName: 'Ilmatar_WS'}),
(cv)-[:hasResult]→(r:Result) RETURN r.value
```

Table F.1
The result of the usability test.

User	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Score
User1	4	3	4	3	5	1	5	3	4	3	72.5
User2	4	3	5	2	4	2	5	2	2	2	72.5
User3	4	2	2	3	4	3	5	3	4	2	65.0
User4	5	2	4	2	4	1	5	1	4	1	87.5
User5	4	1	4	3	4	2	5	2	3	4	70.0
User6	4	1	4	1	5	1	4	1	4	2	87.5
User7	5	2	4	1	5	1	5	1	4	1	92.5
User8	4	2	5	4	4	1	4	2	5	3	75.0
User9	5	1	5	3	5	1	4	1	4	1	90.0
User10	4	2	5	4	4	2	4	2	3	3	67.5
AVG	4.3	1.9	4.2	2.6	4.4	1.5	4.6	1.8	3.7	2.2	78.0

References

[1] Babu MM, Rahman M, Alam A, Dey BL. Exploring big data-driven innovation in the manufacturing sector: evidence from UK firms. Ann Oper Res 2021;1–28.

[2] Wang Y, Ma H-S, Yang J-H, Wang K-S. Industry 4.0: a way from mass customization to mass personalization production. Adv Manuf 2017;5:311–20. http://dx.doi.org/10.1007/s40436-017-0204-7.

[3] Alexopoulos K, Makris S, Xanthakis V, Sipsas K, Chryssolouris G. A concept for context-aware computing in manufacturing: the white goods case. Int J Comput Integr Manuf 2016;29(8):839–49. http://dx.doi.org/10.1080/0951192X.2015.1130257.

[4] Pradeep P, Krishnamoorthy S. The MOM of context-aware systems: A survey. Comput Commun 2019;137:44–69. http://dx.doi.org/10.1016/j.comcom.2019.02.002.

[5] Bisio I, Garibotto C, Grattarola A, Lavagetto F, Sciarrone A. Exploiting context-aware capabilities over the internet of things for industry 4.0 applications. Ieee Netw 2018;32(3):101–7. http://dx.doi.org/10.1109/MNET.2018.1700355.

[6] Elayan H, Aloqaily M, Guizani M. Digital twin for intelligent context-aware IoT healthcare systems. IEEE Internet Things J 2021;8(23):16749–57. http://dx.doi.org/10.1109/JIOT.2021.3051158.

[7] Gil D, Ferrández A, Mora-Mora H, Peral J. Internet of things: A review of surveys based on context aware intelligent services. Sensors 2016;16(7):1069. http://dx.doi.org/10.3390/s16071069.

[8] Firouzi F, Farahani B, Weinberger M, DePace G, Aliee FS. Iot fundamentals: Definitions, architectures, challenges, and promises. In: Intelligent internet of things: From device to fog and cloud. Springer; 2020, p. 3–50.

[9] Jardim-Goncalves R, Grilo A, Popplewell K. Novel strategies for global manufacturing systems interoperability. J Intell Manuf 2016;27:1–9. http://dx.doi.org/10.1007/s10845-014-0948-x.

[10] Brdiczka O, Langet M, Maisonnasse J, Crowley JL. Detecting human behavior models from multimodal observation in a smart home. IEEE Trans Autom Sci Eng 2008;6(4):588–97. http://dx.doi.org/10.1109/TASE.2008.2004965.

[11] Villegas NM, Müller HA. Managing dynamic context to optimize smart interactions and services. In: The smart internet: Current research and future applications. Springer; 2010, p. 289–318.

[12] Abowd GD, Dey AK, Brown PJ, Davies N, Smith M, Steggles P. Towards a better understanding of context and context-awareness. In: Handheld and ubiquitous computing: first international symposium, HUC'99 Karlsruhe, Germany, September 27–29, 1999 proceedings 1. Springer; 1999, p. 304–7. http://dx.doi.org/10.1007/3-540-48157-5_29.

[13] Rico M, Taverna ML, Galli MR, Caliusco ML. Context-aware representation of digital twins' data: The ontology network role. Comput Ind 2023;146:103856. http://dx.doi.org/10.1016/j.compind.2023.103856.

[14] Dey AK. Understanding and using context. Pers Ubiquitous Comput 2001;5:4–7. http://dx.doi.org/10.1007/s007790170019.

[15] Zimmermann A, Lorenz A, Oppermann R. An operational definition of context. In: Modeling and using context: 6th international and interdisciplinary conference, CONTEXT 2007, Roskilde, Denmark, August 20-24, 2007. proceedings 6. Springer; 2007, p. 558–71. http://dx.doi.org/10.1007/978-3-540-74255-5_42.

[16] Hribernik K, Cabri G, Mandreoli F, Mentzas G. Autonomous, context-aware, adaptive Digital Twins—State of the art and roadmap. Comput Ind 2021;133:103508. http://dx.doi.org/10.1016/j.compind.2021.103508.

[17] Baldauf M, Dustdar S, Rosenberg F. A survey on context-aware systems. Int J Ad Hoc Ubiquit Comput 2007;2(4):263–77. http://dx.doi.org/10.1504/IJAHUC.2007.014070.

[18] Zainol Z, Nakata K. Generic context ontology modelling: A review and framework. In: 2010 2nd international conference on computer technology and development. IEEE; 2010, p. 126–30. http://dx.doi.org/10.1109/ICCTD.2010.5646137.

[19] Mahieu C, Ongenae F, De Backere F, Bonte P, De Turck F, Simoens P. Semantics-based platform for context-aware and personalized robot interaction in the internet of robotic things. J Syst Softw 2019;149:138–57. http://dx.doi.org/10.1016/j.jss.2018.11.022.

[20] Rosenberger P, Gerhard D. Context-awareness in industrial applications: definition, classification and use case. Proc CIRP 2018;72:1172–7. http://dx.doi.org/10.1016/j.procir.2018.03.242.

[21] Schilit BN, Theimer MM. Disseminating active map information to mobile hosts. IEEE Netw 1994;8(5):22–32. http://dx.doi.org/10.1109/65.313011.

[22] Lucke D, Constantinescu C, Westkämper E. Smart factory-a step towards the next generation of manufacturing. In: Manufacturing systems and technologies for the new frontier: the 41 st CIRP conference on manufacturing systems May 26–28, 2008, Tokyo, Japan. Springer; 2008, p. 115–8. http://dx.doi.org/10.1007/978-1-84800-267-8_23.

[23] Alegre U, Augusto JC, Clark T. Engineering context-aware systems and applications: A survey. J Syst Softw 2016;117:55–83. http://dx.doi.org/10.1016/j.jss.2016.02.010.

[24] Sipsas K, Alexopoulos K, Xanthakis V, Chryssolouris G. Collaborative maintenance in flow-line manufacturing environments: An industry 4.0 approach. Proc Cirp 2016;55:236–41. http://dx.doi.org/10.1016/j.procir.2016.09.013.

[25] Zhu J, Ong S-K, Nee AY. A context-aware augmented reality assisted maintenance system. Int J Comput Integr Manuf 2015;28(2):213–25. http://dx.doi.org/10.1080/0951192X.2013.874589.

[26] Ringsquandl M, Lamparter S, Lepratti R. Estimating processing times within context-aware manufacturing systems. IFAC-PapersOnLine 2015;48(3):2009–14. http://dx.doi.org/10.1016/j.ifacol.2015.06.383.

[27] Wan G, Dong X, Dong Q, He Y, Zeng P. Context-aware scheduling and control architecture for cyber-physical production systems. J Manuf Syst 2022;62:550–60. http://dx.doi.org/10.1016/j.jmsy.2022.01.008.

[28] Wang P, Liu H, Wang L, Gao RX. Deep learning-based human motion recognition for predictive context-aware human-robot collaboration. CIRP Ann 2018;67(1):17–20. http://dx.doi.org/10.1016/j.cirp.2018.04.066.

[29] Sahlab N, Braun D, Köhler C, Jazdi N, Weyrich M. Extending the intelligent digital twin with a context modeling service: A decision support use case. Proc Cirp 2022;107:463–8. http://dx.doi.org/10.1016/j.procir.2022.05.009.

[30] Grieves M, Vickers J. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In: Transdisciplinary perspectives on complex systems: New findings and approaches. Springer; 2017, p. 85–113.

[31] Zheng Y, Yang S, Cheng H. An application framework of digital twin and its case study. J Ambient Intell Humaniz Comput 2019;10:1141–53. http://dx.doi.org/10.1007/s12652-018-0911-3.

[32] VanDerHorn E, Mahadevan S. Digital Twin: Generalization, characterization and implementation. Decis Support Syst 2021;145:113524. http://dx.doi.org/10.1016/j.dss.2021.113524.

[33] Mylonas G, Kalogeras A, Kalogeras G, Anagnostopoulos C, Alexakos C, Muñoz L. Digital twins from smart manufacturing to smart cities: A survey. Ieee Access 2021;9:143222–49. http://dx.doi.org/10.1109/ACCESS.2021.3120843.

[34] Leng J, Zhang H, Yan D, Liu Q, Chen X, Zhang D. Digital twin-driven manufacturing cyber-physical system for parallel controlling of smart workshop. J Amb Intell Hum Comput 2019;10:1155–66. http://dx.doi.org/10.1007/s12652-018-0881-5.

[35] Liu M, Fang S, Dong H, Xu C. Review of digital twin about concepts, technologies, and industrial applications. J Manuf Syst 2021;58:346–61. http://dx.doi.org/10.1016/j.jmsy.2020.06.017.

[36] Gruber TR. Toward principles for the design of ontologies used for knowledge sharing? Int J Hum-Comput Stud 1995;43(5–6):907–28. http://dx.doi.org/10.1006/ijhc.1995.1081.

[37] Zheng Y, Törmä S, Seppänen O. A shared ontology suite for digital construction workflow. Autom Constr 2021;132:103930. http://dx.doi.org/10.1016/j.autcon.2021.103930.

[38] Hildebrandt C, Köcher A, Küstner C, López-Enríquez C-M, Müller AW, Caesar B, Gundlach CS, Fay A. Ontology building for cyber–physical systems: Application in the manufacturing domain. IEEE Trans Autom Sci Eng 2020;17(3):1266–82. http://dx.doi.org/10.1109/TASE.2020.2991777.

[39] Yang C, Zheng Y, Tu X, Ala-Laurinaho R, Autiosalo J, Seppänen O, Tammi K. Ontology-based knowledge representation of industrial production workflow. Adv Eng Inform 2023;58:102185. http://dx.doi.org/10.1016/j.aei.2023.102185.

[40] Yang C, Zheng Y, Hua Y, Ala-Laurinaho R, Atmojo UD, Tammi K. Knowledge-enhanced digital twin for industrial production process. In: 2024 IEEE 22nd International Conference on Industrial Informatics (INDIN). IEEE; 2024, p. 1–6. http://dx.doi.org/10.1109/INDIN58382.2024.10774266.

[41] Cao J, Vakaj E, Soman RK, Hall DM. Ontology-based manufacturability analysis automation for industrialized construction. Autom Constr 2022;139:104277. http://dx.doi.org/10.1016/j.autcon.2022.104277.

[42] Simon C, Chris L. Time ontology in OWL. 2024, Available online: https://www.w3.org/TR/owl-time/ (accessed on 28 July 2024).

[43] Rasmussen MH, Pauwels P, Lefrançois M, Schneider GF. Building topology ontology. 2024, Available online: https://w3c-lbd-cg.github.io/bot/ (accessed on 28 July 2024).

[44] Janowicz K, Haller A, Cox SJ, Le Phuoc D, Lefrançois M. SOSA: A lightweight ontology for sensors, observations, samples, and actuators. J Web Semant 2019;56:1–10. http://dx.doi.org/10.1016/j.websem.2018.06.003.

[45] Hodgson R, editor. Quantities, Units, Dimensions and Types (QUDT) Schema. 2024, Available online: https://www.qudt.org/doc/DOC_SCHEMA-QUDT.html (accessed on 28 July 2024).

[46] Battle R, Kolas D. Geosparql: enabling a geospatial semantic web. Semant Web J 2011;3(4):355–70.

[47] Arp R, Smith B, Spear AD. Building ontologies with basic formal ontology. Mit Press; 2015, http://dx.doi.org/10.7551/mitpress/9780262527811.001.0001.

[48] ISO Editor. ISO standards maintenance portal. 2023, Available online: https://standards.iso.org/iso-iec/21838/-2/ed-1/en/ (accessed on 22 October 2024).

[49] Mahroof K. A human-centric perspective exploring the readiness towards smart warehousing: The case of a large retail distribution warehouse. Int J Inf Manage 2019;45:176–90. http://dx.doi.org/10.1016/j.ijinfomgt.2018.11.008.

[50] Milazzo M, Ancione G, Brkic VS, Vališ D. Investigation of crane operation safety by analysing main accident causes. In: Risk, reliability and safety: Innovating theory and practice. Taylor & Francis; 2016, p. 74–80.

[51] Milazzo MF, Ancione G, Spasojevic Brkic V, et al. Safety in crane operations: An overview on crane-related accidents. In: Proceedings of the 6th international symposium on industrial engineering, SIE. 2015, p. 36–9.

[52] Beavers JE, Moore J, Rinehart R, Schriver W. Crane-related fatalities in the construction industry. J Constr Eng Manag 2006;132(9):901–10. http://dx.doi.org/10.1061/(ASCE)0733-9364(2006)132:9(901).

[53] ASME B30.2-2022: Overhead And Gantry Cranes. New York, NY: The American Society of Mechanical Engineers, ASME; 2022.

[54] Chen X, He S, Zhang Y, Tong LC, Shang P, Zhou X. Yard crane and AGV scheduling in automated container terminal: A multi-robot task allocation framework. Transp Res C 2020;114:241–71. http://dx.doi.org/10.1016/j.trc.2020.02.012.

[55] Boehning M. Improving safety and efficiency of AGVs at warehouse black spots. In: 2014 IEEE 10th international conference on intelligent computer communication and processing. ICCP, IEEE; 2014, p. 245–9. http://dx.doi.org/10.1109/ICCP.2014.6937004.

[56] Yang C, Tu X, Autiosalo J, Ala-Laurinaho R, Mattila J, Salminen P, Tammi K. Extended reality application framework for a digital-twin-based smart crane. Appl Sci 2022;12(12):6030. http://dx.doi.org/10.3390/app12126030.

[57] Narkhede N, Shapira G, Palino T. Kafka: the definitive guide: real-time data and stream processing at scale. O'Reilly Media, Inc.; 2017.

[58] Jocher G, Chaurasia A, Qiu J. YOLO by ultralytics. 2023, https://github.com/ultralytics/ultralytics (accessed on 28 July 2024).

[59] Knublauch H, Rector A, Stevens R, Wroe C. A Practical Guide to Building OWL Ontologies using the Protégé-OWL Plugin and CO-ODE Tools Edition 1.0. 2004.

[60] Saad M, Zhang Y, Tian J, Jia J. A graph database for life cycle inventory using Neo4j. J Clean Prod 2023;393:136344. http://dx.doi.org/10.1016/j.jclepro.2023.136344.

[61] Bottani E, Vignali G. Augmented reality technology in the manufacturing industry: A review of the last decade. Iise Trans 2019;51(3):284–310. http://dx.doi.org/10.1080/24725854.2018.1493244.

[62] Thorpe S, Fize D, Marlot C. Speed of processing in the human visual system. Nature 1996;381(6582):520–2. http://dx.doi.org/10.1038/381520a0.

[63] Bangor A, Kortum PT, Miller JT. An empirical evaluation of the system usability scale. Int J Hum–Comput Interact 2008;24(6):574–94. http://dx.doi.org/10.1080/10447310802205776.

[64] Kolla SSVK, Lourenço DM, Kumar AA, Plapper P. Retrofitting of legacy machines in the context of Industrial Internet of Things (IIoT). Procedia Comput Sci 2022;200:62–70. http://dx.doi.org/10.1016/j.procs.2022.01.205.

[65] Voigt P, Von dem Bussche A. The eu general data protection regulation (gdpr). In: A practical guide. 1st ed. vol. 10, (3152676):Cham: Springer International Publishing; 2017, p. 10–5555. http://dx.doi.org/10.1007/978-3-319-57959-7.

[66] International Organization for Standardization. Automation systems and integration — Digital twin framework for manufacturing — Part 1: Overview and general principles. ISO 23247-1:2021, Geneva, Switzerland: ISO; 2021, URL https://www.iso.org/standard/75066.html.