



Advancing Industrial Digital Twins: Towards An Open Platform Aligned with Standards

Abdelhadi Belfadel, Stephen Creff, Amira Ben Hamida

► To cite this version:

Abdelhadi Belfadel, Stephen Creff, Amira Ben Hamida. Advancing Industrial Digital Twins: Towards An Open Platform Aligned with Standards. 21st International Conference on Product Lifecycle Management, Jul 2024, Bangkok, Thailand. <hal-04612160>

HAL Id: hal-04612160

<https://hal.science/hal-04612160v1>

Submitted on 21 Aug 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Advancing Industrial Digital Twins: Towards An Open Platform Aligned with Standards

Abdelhadi Belfadel^{1*}[0000–0002–8260–4638], Stephen Creff¹[0000–0002–4690–4668],
and Amira Ben Hamida¹[0009–0001–2394–8385]

IRT SystemX, Palaiseau, France
`firstname.lastname@irt-systemx.fr`

Abstract. Digital Twins are a major driver in the digital transformation of industry. It is gaining in adoption in industry and interest in research and standardization. One of the characteristics of digital twins is the coupling and synchronization between the asset and its virtual representation, raising the issue of interoperability. To date, there is no unique reference architecture for digital twins, nor any implementation covering the full range of capabilities needed to implement digital twins for industrial systems, which may in practice pose a barrier to their adoption and dissemination. Our approach is to move towards an open platform aligned with standards to promote interoperability. The contributions of the paper are twofold: i) a proposal for a high-level architecture of a standards-compliant digital twin platform; ii) a proposal for an open-source implementation on the restricted perimeter of asset/digital twin coupling. This implementation is aligned with the IEC 63278 standard and the Asset Administration Shell vision of Industry 4.0. The article presents 19 criteria applied to the selection of existing open-source implementations, as well as the integration of these into a technical architecture and its implementation.

Keywords: Digital Twin · Industry 4.0 · ISO 23247 · IEC 63278 · Asset Administration Shell · Interoperability

1 Introduction

Digital Twins (DTs) are a major driver in the digital transformation of industry [42]. Gartner identified DTs as one of the “Top 10 Strategic Technology Trends” for the period 2017-2019, e.g. [2]. Besides, MarketsAndMarkets [32] estimates that: “The global digital twin market size in terms of revenue was estimated to be worth USD 10.1 billion in 2023 and is poised to reach USD 110.1 billion by 2028, growing at a CAGR¹ of 61.3% from 2023 to 2028.”, while Gartner [11] states that: “The digital twin market will cross the chasm in 2026 to reach \$183 billion in revenue by 2031”. DTs are therefore gaining in adoption in industry, and have lately been a subject of many interests in different research

* Corresponding author: abdelhadi.belfadel@irt-systemx.fr

¹ Compound Annual Growth Rate

communities [38, 5, 8, 40], and a matter of concern in the field of standardization, e.g. in standardization bodies such as ISO/IEC JTC1/SC41, ISO TC184/SC4, IEC TC65, or consortia such as the Digital Twin Consortium² (DTC), the Industry IoT Consortium³, or the Industrial Digital Twin Association⁴ (IDTA).

Grieves' original description defines a digital twin as a virtual representation of a physical product containing information about the product's characteristics, and has its roots in the field of Product Lifecycle Management (PLM) [14]. From ISO 23247-1 [20], we keep the following definition: a digital twin is a "fit for purpose digital representation [...] of an [...] element with synchronization between the element and its digital representation". We consider the element in a broad sense, i.e. a physical asset, a process, or a system. DTs are complex systems, coupled (interfaced and synchronized) with the physical world. They offer the ability to model real-world entities, processes and systems, and simulate their behavior, while being connected to them. DTs are not objectives in themselves, but means of increasing the operational efficiency of industrial systems. They feature enhanced capabilities for understanding, analyzing, predicting and optimizing the operation and control of the of the real entity [20]. Digital twins enable proactive decision-making and interventions for scenarios like predictive maintenance and logistics optimization. However, they consequently pose challenges such as cyber threats and interoperability issues.

Digital Twin systems are intrinsically complex in their nature and the lack of guidelines makes difficult for corporation to adopt them [1]. To date, there is no unique reference architecture for digital twins, nor any implementation covering the full range of capabilities needed to implement digital twins for industrial systems, which may in practice pose a barrier to their adoption and dissemination. To name few, architectures are defined at different levels: functional and logical reference architecture in ISO 23247-2 [21], or as a reference architecture framework in the upcoming ISO/IEC 30188 ED1 [19], defined by a context diagram and a functional view in ISO/IEC 30173 [23], or at the capacity level and service level [12, 33] by the DTC, or in various articles, as reference architecture e.g. [35, 41], or functional one [5].

Our approach is to move towards an open platform aligned with standards to promote interoperability, defined as: "the degree to which two or more systems, products or components can exchange information and use the information that has been exchanged" [22]. Indeed, the lack of a universal standard poses interoperability challenges, and at the same time we are witnessing the emergence of several proprietary and heterogeneous models, which prevents from the efficient adoption of DTs. Interoperability has to be addressed at the architecture level, and at a data and information one. In that context, the Asset Administration Shell (AAS, specified in IEC 63278 [18]) proposes a standardized electronic representation of industrial assets to enable interoperability.

² <https://www.digitaltwinconsortium.org/>

³ <https://www.iiconsortium.org/>

⁴ <https://industrialdigitaltwin.org/>

The contributions of the paper are twofold: i) a proposal for a high-level architecture of a standards-compliant digital twin platform; ii) a proposal for an open-source implementation on the restricted perimeter of asset/DT coupling. Indeed, this implementation is aligned with the IEC 63278 [18] standard that defines the AAS digital representation coming from Industry 4.0. In addition, we present the criteria applied to the selection of existing open-source implementations, as well as the integration of these into a technical architecture and its implementation.

The remainder of the paper is organized as follows. Section 2 introduces the required capabilities and requirements for a DT. Then, in Section 3, the high-level architecture of a standards-compliant digital twin platform is introduced. Section 4 describes the DT Data Model Building System, considering the criteria to select existing solutions, the evaluation and integration of the final implementation. Later, Section 5 presents some related work. Finally, the last Section draws our conclusion and narrows down possible future works.

2 Digital Twin Capabilities and Requirements

On the one side, Digital twin may have the capabilities of connection, integration, analysis, simulation, visualization, optimization, etc. [23]. These are best described by the DTC. Indeed, as per the DTC, the *Capabilities Periodic Table* (CPT) [12, 33] serves as an architecture- and technology-neutral framework for defining requirements. It targets organizations involved in designing, developing, deploying, and operating digital twins based on capability requirements aligned with specific use cases, as opposed to the features of technology solutions. Following a periodic table approach, the CPT organizes capabilities into clusters based on common characteristics. This structure makes it easily comprehensible in both boardroom discussions for securing funding for digital twin projects and on the shop floor when gathering requirements for digital twin applications. The CPT offers visual guidance to facilitate collaboration, brainstorming, and the explicit articulation of capability requirements.

In this context, capability refers to the capacity to perform specific actions or achieve desired outcomes. The CPT specifically focuses on technology-based capabilities while remaining agnostic to particular technology or product solutions. The capabilities of digital twins are broken into six key high-level categories: *CPT1: data services*, *CPT2: integration*, *CPT3: intelligence*, *CPT4: user experience (UX)*, *CPT5: management* and *CPT6: trustworthiness*. Each of these categories further expands into 62 discrete top-level capabilities.

On the other side, regarding requirements enabling DT platform, several contributions such as in [10] offers an initial synthesis of DT requirements, identified through a literature review and industry interviews. The highlighted concerns include real-time data, integration, fidelity, interaction, communication, convergence, automation, updates, autonomy, connectivity, data-related aspects, efficiency, expansibility, and scalability. This integration results in a comprehensive

set of functional and non-functional requirements as compiled by [34]. These requirements are related to interoperability of the DT, with the assets, or other services.

The contribution made by [25] introduced a set of requirements that can also be relevant to our broader context, facilitating the safe and harmonious operation of our digital twin platform alongside its physical counterpart.

Considering standardization bodies, in 2021, the ISO launched efforts with the release of the four parts ISO 23247 series for DTs in manufacturing. Part 1 [20] is related to requirements, and is concerned with some functional requirements (Communication, Data acquisition, Data analysis, Identification, Management, Product life-cycle, Simulation, Synchronization, Viewpoint, Hierarchical modelling of digital twin for manufacturing), and some “-ilities” (Accuracy, Data integrity, Extensibility, Granularity, Security).

These requirements are considered when building the architecture described in the following section.

3 High-level Digital Twin platform functional architecture

ISO 23247 Part 2 [21] of the series defines a reference architecture for DTs in the manufacturing domain, that includes (i) an entity-based reference model, and (ii) a functional view of the entity-based reference model with specified functional entities. The entity-based reference model uses a segmented layered pattern with sidecar [3], each of which has its own specified process logic:

The Device Communication entity (23247-DCE): in charge of for collecting data from the Observable Manufacturing Elements (OMEs) - including personnel, equipment, materials, manufacturing processes, facilities, environment, products, and supporting documents - and controlling the OMEs;

The Digital Twin entity (23247-DTE): responsible for modelling the data collected from the Device Communication entity and providing functionalities such as realization, management, synchronization, and simulation. The digital twin entity digitally represents OMEs;

The User entity (23247-UE): that host the application of the framework;

The Cross-System entity (23247-CSE): that resides across all the other entities to provide common functionalities such as data translation, data assurance, and security support.

Following a component architecture pattern, we define a high-level functional architecture aligned with the ISO 23247-2. The proposed high-level architecture, cf. Figure 1, offers a comprehensive overview of the technology foundations, ensuring a thorough understanding of the system’s structure. It is important to note that interactions between these components will occur either through Application Programming Interfaces (APIs) or direct calls within each component. This architectural representation has been deliberately kept at a high level of abstraction to accommodate page limits and maintain consistency across the paper.

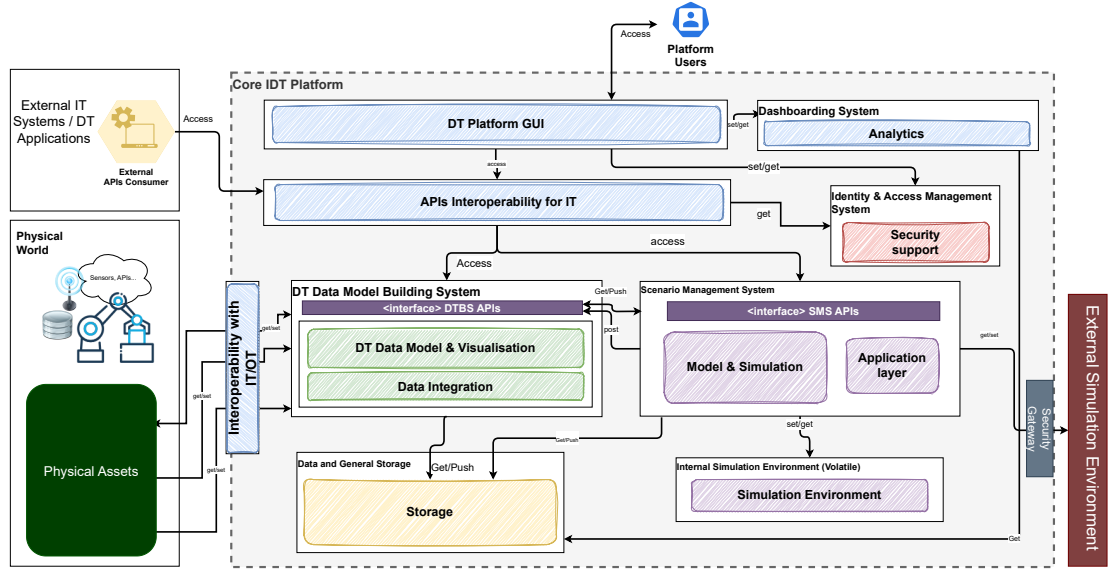


Fig. 1: DT Platform High-level Functional Decomposition Architecture Diagram

- **Platform UI:** This user-friendly dashboard provides customizable interfaces tailored to each platform actor, facilitating dynamic what-if scenarios runs and result exploration in a workflow-driven manner. Users can create DT data models, what-if scenarios with specific metadata or Key Performance Indicators (KPIs), import or design data models, and define threshold parameters. The interface orchestrates simulation scenarios as a pipeline of models (simulation models, AI models, data transformers, etc.), with KPI metrics displayed for both the physical and digital worlds. This component is mapped to *23247-UE* and *CPT4*.
- **Analytics:** A dashboarding component that collects from the data and general storage of the platform, monitors, and displays complex data via interactive and customizable dashboards and reports. This component is mapped to *23247-UE* and *CPT3*.
- **APIs Interoperability for IT:** This component plays a crucial role in seamlessly integrating the digital twin platform with external IT systems. By exposing its APIs, it enables external DT applications to effortlessly consume platform features, facilitating enhanced interoperability and streamlined communication between systems. Additionally, it oversees essential security measures and administrative tasks such as authentication, metric collection, input validation, and response transformation, ensuring the smooth and secure operation of the integrated ecosystem. This component is part of *23247-UE* and *CPT2*.
- **DT Data Model Manager (DTDMM) & Visualization:** This component manages digital twin data models, updating them in the database

based on synchronization from physical objects or decisions from the decision system. It also offers real-time visualization of the digital twin replica. This component is part of *23247-DCE* and *CPT1*.

- **Data Integration:** This component synchronizes data from external Information Systems and physical assets, forwarding updates to the Data Model Manager for persistence in the appropriate DT Data Model instance. This component is part of *23247-DCE* and *CPT2*.
- **Security Support:** This vital component ensures strict access control to safeguard an organization’s data and resources. It oversees authentication processes, user management, role assignments, service access permissions, and DT data model consumption rights, guaranteeing that only authorized individuals can access sensitive information and perform designated actions. This component is part of *23247-CSE* and *CPT6*.
- **Application Layer:** This component empowers users to design workflow scenarios using registered AI or simulation models and facilitates the deployment and initiation of simulation environments. This component is part of *23247-DTE* and *CPT3*.
- **Modeling & Simulation:** This component integrates model registration, workflow execution, and decision support functionalities. It enables the design and execution of workflow scenarios, aids decision-making based on simulation outcomes and predefined KPIs, and seamlessly executes workflows in external environments. This component is part of *23247-DTE* and *CPT3*.
- **Data and General Storage:** This component handles diverse data storage needs within the platform, offering multiple APIs for access to various storage types tailored to each component’s specific requirements and constraints. This component is part of *23247-DTE* and *CPT1*.

4 DT Data Model Building System

The growing necessity for the adoption and seamless creation, processing, and integration of engineering digital twins emphasizes the importance of adhering to standards. Considering the DT Data Model Building System, the latest standard published by IEC 63278-1:2023 [18] specifies the configuration of a standardized digital representation of an asset, referred to as the AAS. The AAS ensures consistent access to information and services. Its primary objective is to facilitate the secure and trustworthy exchange of information between two or more software applications, allowing them to mutually utilize the exchanged information.

4.1 Overview of the Functional Analysis

The functional analysis per each component is made from three perspectives. *Behavior and Functionality* detailing a story map [36] outlining features and associated user stories, *Interaction Descriptions* which outline interactions between components and users, and *UI mock-ups and sequence diagrams* providing

insight into component interactions with users or other platform components as necessary.

In terms of the behavior and functionality, story maps describe the functionality on different levels of aggregation/abstraction. User stories have been defined for the two components of the *DT Data Model Building System*: 1) For the *DT Data Model Manager* (DTDMM): create, update, delete, list, import, and export data model; 2) For the *Data Integration Manager*: (From physical asset to digital asset) receive data updates from the IT/OT, (From physical asset to digital asset) Send data to DTDMM, (From digital asset to physical asset) Receive DM updates from DTDMM, (From digital asset to physical asset) Send update request to the IT/OT; and Data transformation from DTDMM to send to the IT/OT, and Data transformation to send to DTDMM.

4.2 Criteria for AAS open-source implementations selection

Several prior contributions, such as those proposed by [24] and [26], presented an overview of existing AAS implementations with the goal of identifying potential candidates for implementing our targeted AAS-based DT platform. However, new contributions have emerged in GitHub repositories, and updated versions have been released since then. In order to conduct a thorough review and improvement of the identified open-source solutions, aiming to select those that comprehensively cover the functional aspects of the AAS specification, we define two types of criteria functional and technical ones, listed below.

Functional criteria:

- **F1:** Creating/Generating a Submodel Instance from an Established Submodel Template/Type (publicly available or from a manufacturer’s specification).
- **F2:** Concept description: It refers to the ability to describe (or link to) the semantics of a property
- **F3:** Offering Interfaces and APIs for Accessing AAS Information: Functionalities, operations, and methods of the Administration Shell can be made accessible through an API.
- **F4:** Registry implementation: Infrastructure that hosts and interconnects multiple AASs together (implement registry services, discovery services, end-point handling, and other related functionalities).
- **F5:** Event Management (In/Out): Events are a very versatile mechanism of the AAS. The different types are: Forward events; Reverse events; Tracking events and Value push events across clouds.
- **F6:** AAS Security Management: It involves establishing access control mechanisms and defining access permission rules that specify the permissions granted to authenticated subjects for accessing specific objects.
- **F7:** AASX file formats: Management of file formats to exchange the structure of the AAS (incl. associated values and persistence management).
- **F8:** Filtering Import/Export Information.
- **F9:** Facilitating Information Sharing Across Various Systems within the scope of the entire RAMI4.0. Various data formats, such as OPC-UA (IEC62541),

AutomationML (IEC 62714), XML (ISO 8879) / JSON (ISO/IEC 21778), and Resource Description Framework, are utilized or recommended for use in different product life-cycle phases.

- **F10:** Referencing in AAS: Within the context of AAS, two distinct types of references are identified: *Global references* (unique identifiers, possibly concatenated from multiple sources) and *Model references* (ordered keys, enabling precise element access via concatenated paths).

Technical criteria:

- **T1:** Modeling language: Technical approach to systems engineering that focuses on the creation and operation of the AAS model. Several paradigms might be used such as Object-Oriented Programming (OOP), Model-based systems engineering (MBSE), Business Process Modeling Notation (BPMN), Service-Oriented Architecture (SOA) and others.
- **T2:** Communication protocol: For example OPC-UA, REST, MQTT (ISO/IEC 20922), MTConnect (ANSI/MTC1.4), CoAP (RFC 7252), SOAP (W3C), AMQP (ISO/IEC 19464), etc.
- **T3:** ASS Registry: the AASX Registry stores registry information from AASX Servers and other AAS servers. It allows to find AAS endpoints by AssetIDs.
- **T4:** Data storage: the distributed AAS approach allows for utilizing AAS and Submodels as decentralized data storage and a unified interface to data sources. This means static data, databases, and tools can be outfitted with a Submodel interface to offer different types of data as AAS Submodels.
- **T5:** Type of solution: Can be a Software Development Kit (SDK), a Docker container, a JAR file, or a library.
- **T6:** Containerization: simplifies software deployment by packaging an application's code and dependencies into a single (OS independent) container.
- **T7:** License and Copyright type.
- **T8:** Documentation: Comprehensive, appropriate, well-structured user documentation.
- **T9:** User interface: Does the application offer a user interface?

4.3 Evaluation of existing solutions

The assessment of most criteria has been conducted via documentation review, application deployment, and rigorous testing, or through a comprehensive analysis of the codebase, leading to Table 1. Based on our assessment, the optimal solutions for each category, enabling the development of a cloud-based DT platform and adhering to the specified criteria, standards and constraints, are as follows:

- **Back-end solution:** Basyx Java Server SDK V2⁵, implementing both Part 1 and Part 2 specifications of AAS V3.
- **Front-end solution:** AAS Web Client (Faunhover AAS Graphical User Interface - GUI).⁶

⁵ <https://github.com/eclipse-basyx/basyx-java-server-sdk>

⁶ <https://github.com/eclipse-basyx/basyx-applications/tree/main/aas-gui>

Solution	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	T1	T2	T3	T4	T5	T6	T7	T8	T9
BaSyx SDK V1 (aas spec. v2.0)	X		X	X	(X)	X	X			X	OOP (Java), BPMN engine (camunda), SOA	OPC UA, HTTP REST, MQTT, Basyx native TCP/UDP, JSON serialization supports ActiveMQ or Kafka	X	InMemory	SDK, Compliant with Details of the AAS Part 1 V2.0.1	Maven, JAR	Eclipse Public License v 2.0	X	
BaSyx SDK V2 (aas spec. v3.0)	X	X	X	X	(X)	X	X		(X) interface with DataBridge Component	X	OOP (Java)	HTTP(s), MQTT, Kafka	X	InMemory (by default), MongoDB	SDK	Maven Jar, Docker	MIT Licence	X	
AAS Server	X	X	X	X	(X)	X	X		X	X	OOP (Java)	HTTP, MQTT		InMemory (by default), MongoDB	Backend Web App	Docker	MIT Licence	X	
AAS DataBridge Component					X			X	X		OOP (Java), SOA	OPC UA, HTTP REST, MQTT, ActiveMQ, Kafka		InMemory; File-based; Will support database storage in version 1.0.0	Backend Web App	Docker	Apache License V2.0	X	
AAS Web Client (AAS GUI)	X						X					HTTP REST	X		Web App GUI	Docker	MIT Licence	X	X
FAAAST-Service	X	X	X		(X)	X	X	X	(X) OPC UA, aasx...		OOP (java)	HTTP(S), OPC UA, MQTT		InMemory; File-based; will support database storage in version 1.0.0	Web App (Backend)	Docker	Apache License Version 2.0	X	
Py140AAS	X						X			X	OOP (python)			In-memory; CouchDB;	SDK		MIT License	X	
Aasx-server	X	X	X		X	X	X	X	X (HTTP(S) OPC UA, MQTT, RDF, XML, XMI)	X		HTTP(S); OPC UA	X	File-based	Web App (Backend)	Docker	Apache License 2.0	X	X
Aasx-package-explorer	X	X	X		X	X	X	X	X (HTTP(S), OPC UA, MQTT, File with different types (json, xml, aml, aasx...))	X	JSON; XML; AASX files; AutomationML (AML);	HTTP(S); OPC UA; MQTT		File-based	Windows Software	Binaries package for Windows 10	Apache License 2.0	X	X
NOVAAS	X				X	X	X	X	X (HTTP, MQTT; TCP socket connections)		Low-code development platform (LCDP)	HTTP; MQTT; OPCUA; TCP		file-based; InMemory; SQLite	Web App	Docker	European Union Public License 1.2	(X)	X

Table 1: Tools analysis and evaluations

– **Data integration solution:** Basyx Data Bridge component.⁷

The Basyx Java Server SDK V2 back-end presents numerous advantages. It is constructed on the AAS4J data model library, which is accessible in the Eclipse project repository. This library adheres to the latest AAS V3 specification release and complies with IEC 63278 [18]. This solution seamlessly integrates multiple independent components, encompassing an AAS Repository server, an AAS Registry server, a Submodel Repository server, and a Submodel Registry server.

Some noteworthy features include the ability to deploy multiple server instances for scalability, with each instance exposing its own APIs and Swagger interface. It also simplifies model storage in a MongoDB database, utilizing memory storage by default, and supports event communication through Kafka or MQTT protocols. However, it is essential to highlight a potential limitation:

⁷ <https://github.com/eclipse-basyx/basyx-databridge>

the incorporation of the Basyx data bridge component is mandatory for data integration.

4.4 Platform Implementation and Integration

In addition to the major integration effort of different components, it is important to mention some functional improvements done comparing to the state of the art such as: an automatic registration of an AAS model to the registry, and the possibility to import AAS file in different format for DT model instantiation in our platform. Derived from the solutions chosen in the analysis stage, Figure 2 illustrates the technical architecture for this initial platform release ⁸.

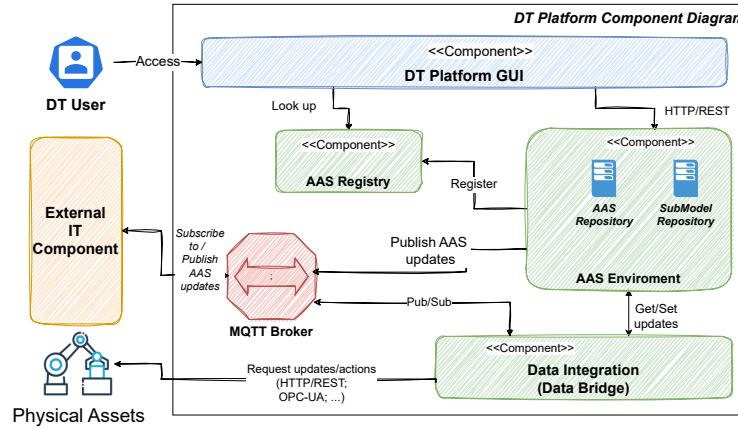


Fig. 2: Industrial DT Integrated Platform - Component Diagram

The AAS server component is used to host several AAS and Submodels. It provides access to reactive AAS and Submodel instances through an AAS API. In this version of the platform, Submodel instances can be serialized separately and can be hosted on separate AAS server applications, which can be also accessed by the AAS API (a deployment of a Submodel registry is needed in this case). Each AAS Server Application and each Submodel may have an endpoint for the API access. Such an endpoint defines an address to access the AAS Server Application by the AAS API specified protocols and operations. In addition, a registry specification is developed to register endpoints and descriptions for AAS Server Applications. Such AAS Registry contains Endpoints and IDs of AAS instance and IDs of Submodel instance from a related AAS server application. These entries can be found by requesting the registry.

The Data Integration component, facilitated by the Data Bridge SDK, supports seamless integration with various protocols including OPC UA, HTTP

⁸ The source code will be available at this public repository <https://github.com/IRT-SystemX/jni-platform>

REST, MQTT, ActiveMQ, and Kafka. It not only enables data transmission but also empowers transformation across these protocols. This component embraces two integration patterns: On-demand delegation, where data is queried, transformed, and returned upon request by the AAS server (pull); and cyclic integration, where data is periodically queried, transformed, and integrated into the AAS or triggered by events (push).

Interactions within the system outline the sequence of operations executed by client applications to accomplish specific goals in various use cases. In upcoming iterations of this platform, users will gain the ability to craft workflows tailored to what-if scenarios, thereby augmenting functionality and expanding their capabilities.

During platform UI enhancements, a new feature has been introduced to facilitate the importation of AAS files. Additionally, several updates have been added to the reused AAS GUI to comply with the latest V3 AAS specification and dynamically retrieve AAS from discovered AAS Repository endpoints and gather information about their Submodel references. Leveraging these references, along with a set Submodel registry and distributed architecture deployment, the UI communicates with the Submodel Registry for respective Submodel Descriptors and sends requests to the Submodel Registry, then to the Submodel repository interface based on defined endpoints.

Figure 3 showcases the integrated GUI providing access to platform features. Furthermore, a reactive AAS data model, along with its corresponding sub-models, is instantiated for a real industrial energy-oriented use case.

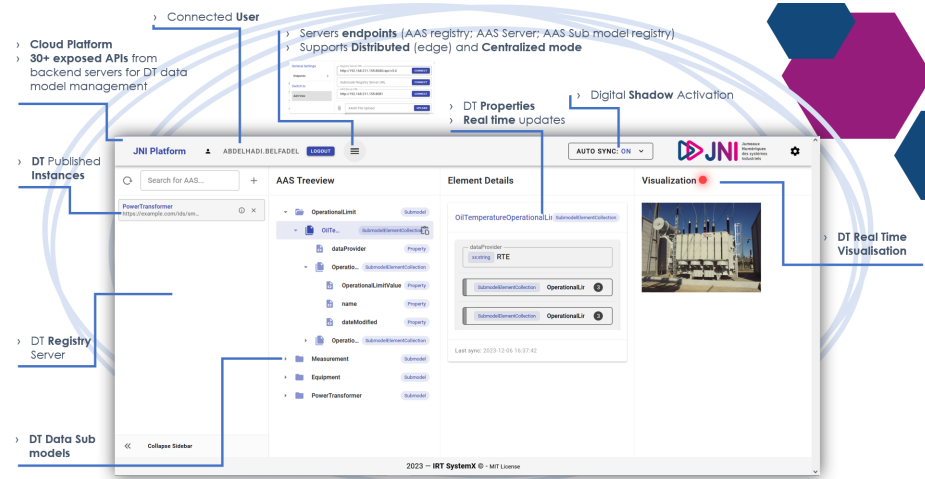


Fig. 3: Industrial DT Integrated Platform - GUI and an AAS instance displayed

5 Related Works

In Industry 4.0, smart manufacturing techniques like dynamic modeling and simulation, coupled with Digital Twin (DT) technology, are enhancing production capabilities by mirroring physical systems virtually, facilitating real-time communication between cyber and physical spaces for improved manufacturing efficiency [9, 39, 29, 17]. It fosters continuous improvement towards human well-being and overall quality of life [9]. Additionally, DT unlocks new possibilities for Industry 4.0, making it a strategic priority for leading organizations. Leveraging DT simulations and optimized decision-making, enterprises develop self-aware smart products, gaining competitive advantages [37, 13]. With benefits spanning product design to maintenance [31], DT also plays a pivotal role in developing smart machine tools [43] and has potential as a cloud service, shaping future technologies.

In this regard, authors in [28] outline the technology stack for DT establishment. They start with the Service layer, detailing services for shop-floor assets and modeling. Next, the Data Computation layer covers sensor data querying, aggregation, and processing. Data Representation, storage tools, and machine learning methodologies are summarized. The Data Management and Connectivity layer examines tools for data acquisition and transmission.

Regarding DT perspectives in PLM stages, [28] analyzes them using a generic competitive process framework proposed by [7], outlining DT's implementation structure and process. This framework allows companies to focus on relevant PLM aspects for product enhancement. The contributions identified involve engineering PLM phases and highlight various industry applications. For instance, the Cyber Guided Vehicle in M2M logic project [4] utilizes a Discrete Event Simulation (DES) approach with sensors primarily focused on proximity detection. Similarly, the Framework for CPPS-DT implementation [16] also employs DES and utilizes communication protocols like MQTT, OPC, and WebSocket. Conversely, the Multi-dimensional DT for prestressed steel project [30] relies on simulation tools such as ABAQUS and ANSYS, employing serial communications with sensors measuring cable force and displacement. Lastly, the Digital Twins on High Voltage Transmission Line Live Working Scenes project [15] uses a Static Electrical Field Finite Element (FE) approach and incorporates sensors such as accelerometers, gyroscopes, and magnetic field sensors.

Finally, in the domain of cybersecurity, [6] assesses the current threat landscape concerning industrial control systems communicating with the AAS, as well as IT systems hosting the AAS. The authors evaluate the relevance of these threats to the AAS and delve into detailed analyses of the most pertinent threats, specifically Basic Web Application Attacks and Malware Infections. Comparisons of recommended countermeasures for these threats with those outlined in BSI [6] and IEC 62443-4-2 [27] reveal gaps and missing countermeasures, highlighting areas for further research to bolster the overall security of the AAS.

6 Conclusion and Future Work

We presented in this contribution i) a high-level architecture of a standards-compliant digital twin platform (ISO 23247), and ii) an industrial DT platform that is designed in alignment with IEC 63278. With a primary goal to facilitate the design of industrial digital twin systems for various use cases. The produced open platform integrates and enhances existing open-source tools. This strategic approach aims to accelerate the implementation of digital twins, fostering their widespread adoption within the industry while ensuring compliance with industry standards and cybersecurity protocols.

The ongoing development efforts prioritize several critical aspects, including the adoption of a risk-based analysis approach, which facilitates a cyber-by-design approach to support the life-cycle of Digital Twins and ensure cybersecurity assurance. In addition, key components such as Identity & Access Management, Scenario Management System for what-if scenarios and Decision System are under conception and development for future releases. These components aim to enhance security, streamline workflow execution, assist decision-making processes, and optimize resource utilization within the integrated industrial Digital Twins platform. Their implementation represents significant strides towards achieving greater efficiency, effectiveness, and adaptability in industrial operations.

Acknowledgements. This work has been supported by the French government under the “France 2030” program, as part of the SystemX Technological Research Institute.

References

1. Ahmad, B., Harrison, R., Vera, D.: A connective framework to support the lifecycle of cyber-physical production systems. *Proceedings of the IEEE* **109** (02 2021). <https://doi.org/10.1109/JPROC.2020.3046525>
2. Analyst(s): Marc Halpern, Simon Jacobson, R.F.: Hype cycle for it evolution in manufacturing, id: G00340295. Tech. rep., Gartner (2018)
3. Bass, L.J., Clements, P.C., Kazman, R.: *Software Architecture in Practice*. SEI series in software engineering, Addison-Wesley-Longman (1999)
4. Bottani, E., Cammardella, A., Murino, T., Vespoli, S., et al.: From the cyber-physical system to the digital twin: the process development for behaviour modelling of a cyber guided vehicle in m2m logic. XXII Summer School Francesco TurcoIndustrial Systems Engineering pp. 1–7 (2017)
5. Boyes, H., Watson, T.: Digital twins: An analysis framework and open issues. *Computers in Industry* **143**, 103763 (2022). <https://doi.org/10.1016/j.compind.2022.103763>
6. Bröring, A., Ehrlich, M., Trsek, H., Wisniewski, L.: Secure usage of asset administration shells: an overview and analysis of best practises. <https://opendata.uni-halle.de/handle/1981185920/41505> (2021)
7. Casadesus-Masanell, R., Ricart, J.E.: From strategy to business models and onto tactics. *Long range planning* **43**(2-3), 195–215 (2010)

8. Dalibor, M., Jansen, N., Rumpe, B., Schmalzing, D., Wachtmeister, L., Wimmer, M., Wortmann, A.: A cross-domain systematic mapping study on software engineering for digital twins. *Journal of Systems and Software* **193**, 111361 (2022). <https://doi.org/10.1016/j.jss.2022.111361>
9. Davis, J., Edgar, T., Porter, J., Bernaden, J., Sarli, M.: Smart manufacturing, manufacturing intelligence and demand-dynamic performance. *Computers & Chemical Engineering* **47**, 145–156 (2012)
10. Durão, L.F.C., Haag, S., Anderl, R., Schützer, K., Zancul, E.: Digital twin requirements in the context of industry 4.0. In: *Product Lifecycle Management to Support Industry 4.0: 15th IFIP WG 5.1 International Conference, PLM 2018, Turin, Italy, July 2-4, 2018, Proceedings 15*. pp. 204–214. Springer (2018)
11. Gartner: Emerging technologies: Revenue opportunity projection of digital twins (2022), <https://www.gartner.com/en/documents/4011590>, accessed on Feb. 26, 2024
12. Geater, J., Hirsch, F., Richter, D., Robkin, M., Zahavi, R.: Iot security maturity model digital twin profile - an industry iot consortium and digital twin consortium whitepaper. Tech. rep., DTC (6 2022)
13. Ghobakhloo, M.: The future of manufacturing industry: a strategic roadmap toward industry 4.0. *Journal of manufacturing technology management* **29**(6), 910–936 (2018)
14. Grieves, M.: Digital twin: manufacturing excellence through virtual factory replication. White paper **1**(2014), 1–7 (2014)
15. Guo, J., Wu, Q., Guo, K., Xiong, S., Feng, W., Xue, J.: Study on the construction and application of digital twins on high voltage transmission line live working scenes. *IEEE Access* **9**, 111587–111594 (2021)
16. Harrison, R., Vera, D.A., Ahmad, B.: A connective framework to support the lifecycle of cyber-physical production systems. *Proceedings of the IEEE* **109**(4), 568–581 (2021)
17. I-Scoop: Digital twin technology and simulation: Benefits, usage and predictions 2018. pp. 1–6 (November 11 2017)
18. IEC: Asset administration shell for industrial applications - part 1: Asset administration shell structure
19. IEC: Iso/iec 30188 ed1 - digital twin - reference architecture (2024), https://www.iec.ch/dyn/www/f?p=103:38:793266745535:::FSP_ORG_ID,FSP_APEX_PAGE,FSP_PROJECT_ID:20486,23,104896, accessed on Feb. 26, 2024
20. ISO 23247-1:2021 - Automation systems and integration Digital twin framework for manufacturing - Part 1: Overview and general principles. Standard, International Organization for Standardization, Geneva, CH (Oct 2021)
21. ISO 23247-2:2022 - Automation systems and integration Digital twin framework for manufacturing — Part 2: Reference architecture. Standard, International Organization for Standardization, Geneva, CH (Oct 2021)
22. ISO/IEC 25010:2023 - Systems and software Quality Requirements and Evaluation (SQuaRE) Product quality model. Standard, ISO/IEC, Geneva, CH (Nov 2023)
23. ISO/IEC 30173:2023 - Digital twin – Concepts and terminology. Standard, ISO/IEC, Geneva, CH (Nov 2023)
24. Jacoby, M., Baumann, M., Bischoff, T., Mees, H., Müller, J., Stojanovic, L., Volz, F.: Open-source implementations of the reactive asset administration shell: A survey. *Sensors* **23**(11), 5229 (2023)
25. Kasper, L., Birkelbach, F., Schwarzmayr, P., Steindl, G., Ramsauer, D., Hofmann, R.: Toward a practical digital twin platform tailored to the requirements of industrial energy systems. *Applied Sciences* **12**(14), 6981 (2022)

26. Kilian, L.: Implementing an Asset Administration Shell based on Open-Source Frameworks. Bachelor's thesis (2023)
27. Kronfuss, E.: Industrial cyber security standard-iec 62443. Tech. rep. (2018)
28. Lim, K.Y.H., Zheng, P., Chen, C.H.: A state-of-the-art survey of digital twin: techniques, engineering product lifecycle management and business innovation perspectives. *Journal of Intelligent Manufacturing* **31**, 1313–1337 (2020)
29. Liu, J., Zhou, H., Liu, X., Tian, G., Wu, M., Cao, L., Wang, W.: Dynamic evaluation method of machining process planning based on digital twin. *IEEE Access* **7**, 19312–19323 (2019)
30. Liu, Z., Bai, W., Du, X., Zhang, A., Xing, Z., Jiang, A.: Digital twin-based safety evaluation of prestressed steel structure. *Advances in Civil Eng.* pp. 1–10 (2020)
31. Mabkhot, M.M., Al-Ahmari, A.M., Salah, B., Alkhalefah, H.: Requirements of the smart factory system: A survey and perspective. *Machines* **6**(2), 23 (2018)
32. MarketsAndMarkets: Digital twin market size, share, statistics and industry growth analysis report by application [...], industry [...], enterprise and geography - global growth driver and industry forecast to 2028 (2022), <https://www.marketsandmarkets.com/Market-Reports/digital-twin-market-225269522.html>, accessed on Feb. 26, 2024
33. McKee, D.: Platform stack architectural framework: An introductory guide - a digital twin consortium whitepaper. Tech. rep., DTC (7 2023)
34. Moyne, J., Qamsane, Y., Balta, E.C., Kovalenko, I., Faris, J., Barton, K., Tilbury, D.M.: A requirements driven digital twin framework: Specification and opportunities. *IEEE Access* **8**, 107781–107801 (2020)
35. Newrzella, S.R., Franklin, D.W., Haider, S.: Three-dimension digital twin reference architecture model for functionality, dependability, and life cycle development across industries. *IEEE Access* **10**, 95390–95410 (2022). <https://doi.org/10.1109/ACCESS.2022.3202941>
36. O'hEocha, C., Conboy, K.: The role of the user story agile practice in innovation. In: *International Conference on Lean Enterprise Software and Systems*. pp. 20–30. Springer (2010)
37. Posada, J., Toro, C., Barandiaran, I., Oyarzun, D., Stricker, D., De Amicis, R., Pinto, E.B., Eisert, P., Döllner, J., Vallarino, I.: Visual computing as a key enabling technology for industrie 4.0 and industrial internet. *IEEE computer graphics and applications* **35**(2), 26–40 (2015)
38. Qi, Q., Tao, F., Hu, T., Anwer, N., Liu, A., Wei, Y., Wang, L., Nee, A.: Enabling technologies and tools for digital twin. *Journal of Manufacturing Systems* **58**, 3–21 (2021). <https://doi.org/10.1016/j.jmsy.2019.10.001>
39. Schneider, G.F., Wicaksono, H., Ovtcharova, J.: Virtual engineering of cyber-physical automation systems: The case of control logic. *Advanced engineering informatics* **39**, 127–143 (2019)
40. Segovia, M., Garcia-Alfaro, J.: Design, modeling and implementation of digital twins. *Sensors* **22**(14) (2022). <https://doi.org/10.3390/s22145396>
41. Tao, F., Sun, X., Cheng, J., Zhu, Y., Liu, W., Wang, Y., Xu, H., Hu, T., Liu, X., Liu, T., Sun, Z., Xu, J., Bao, J., Xiang, F., Jin, X.: maketwin: A reference architecture for digital twin software platform. *Chinese Journal of Aeronautics* **37**(1), 1–18 (2024). <https://doi.org/10.1016/j.cja.2023.05.002>
42. Tao, F., Zhang, H., Liu, A., Nee, A.Y.C.: Digital twin in industry: State-of-the-art. *IEEE Transactions on Industrial Informatics* **15**(4), 2405–2415 (2019). <https://doi.org/10.1109/TII.2018.2873186>
43. Xu, X.: Machine tool 4.0 for the new era of manufacturing. *The International Journal of Advanced Manufacturing Technology* **92**, 1893–1900 (2017)