Case Study:

Digitizing the Process of Obtaining the Certification of Aptitude for Service (CAS) of Locomotives in the Spanish Railway Network

By the DAFMM IMA Research Committee

January 2025

International Maintenance Association (IMA)
Via dei Gilardi 15 6926
Montagnola Lugano, **Switzerland**
Tel: (41) 91 225 87 13 | Email: info@ima-world.org
Website: www.ima-world.org

CONTENT

1. EXECUTIVE SUMMARY

1.1. Purpose and Context

The study demonstrates the step-by-step implementation of digitalization strategies for a railway maintenance process, aligned with the recommendations for structuring data for digital assets described in the document entitled: "Digitalization of Assets, Facilities and Maintenance Management" as presented by the DAFMM IMA Research Committee. This includes:

- Following a maturity matrix framework for assessing and enhancing digital and maintenance management capabilities.
- Incorporating Digital Twins and IoT-based ecosystems for real-time monitoring and data integration.
- Leveraging Asset Data Models (Asset Definition, Criticality, Monitoring, and Intelligent Management Models).

1.2. Key Highlights

1. Initial Assessment and Objectives:
   - Evaluated the workshop's current digital maturity, identifying a low starting point with basic paper-based processes.
   - Defined objectives to digitize the CAS process (the process to Certificate the locomotive for its Aptitude for the Service) using advanced data models and technologies to achieve Level 4 Digital Maturity.
2. Data and Technology Integration:
   - Applied the Asset Definition Model to codify machine components and systems for seamless digital representation.
   - Integrated IoT systems for real-time data acquisition and automated workflows, enhancing efficiency and traceability.
   - Developed a Data Lakehouse Architecture to store, process, and analyze data for predictive and condition-based maintenance.
3. Digitalization Implementation:
   - Established Digital Twin technology to simulate and monitor locomotive systems and workshop operations.
   - Developed ETL (Extract, Transform, Load) pipelines for data transformation and integration, enabling predictive analytics.
4. Training and Scalability:
   - Trained the workforce in using digital tools, fostering alignment with advanced maintenance management frameworks.
   - Piloted the solution in a controlled environment before scaling across other workshop operations.

1.3. Strategic Significance

This case demonstrates how the structured digitalization framework outlined in the IMA Research Committee document can be operationalized, providing a replicable model for transforming maintenance processes in compliance with industry standards and best practices.

Figure 1. Digitalizing the CAS Process

By digitizing CAS certification (Figure 1), the case study establishes a foundation for broader maintenance management digitalization, illustrating how data models and digital twins can align with operational and strategic goals in asset-intensive industries.

2. INTRODUCTION TO THE WORK

The following case study – digitalization of the process of obtaining the Certificate of Aptitude for Service – shows how to digitalize a process. The asset's data models are defined according to structuring data for digital assets defined in this document. And it is proposed to follow the step-by-step guide for digitalizing Asset, Facilities and Maintenance Management as included in the IMA Research Committee document to this respect. This case study is part of a real project aimed at developing a digital twin.

The project aims to digitally transform the rail transport sector. The digitalization of the rail infrastructure maintenance service is carried out. The aim is to investigate the design, development and use of digital twin models for maintenance services. Now only one process included in the twin will be digitized.

Initially, it was decided to digitize the process of obtaining the CAS of the machinery fleet of a company responsible for the maintenance of the railway tracks in Spain. This document aligns with the recommendations included in steps 1 to 4 of the IMA document, where data models, architectures, and platforms for digitalization are detailed. The following points outline the steps to digitize this process, serving as the foundation for the future digitalization of all maintenance management:

1. *Assess the Current State of Maintenance Management:* Assessment of the current process of obtaining the CAS. Identifying the technology used, the data framework and the level of digitalization of the process.
2. *Define Digitalization Goals and Objectives.* Define the objectives to be achieved with the digitalization of this process.
3. *Select and Implement Digital Technologies.* Establish robust data pipelines using ETL (Extract, Transform, Load) models for seamless data integration using specific platforms. Implementation de la technology de Digital Twins.
4. Build the data infrastructure: Development of the Data Lakehouse platform.

5. Train and Upskill the Workforce. Provide training on the use the new tool for obtaining the CAS.
6. *Pilot and Test Digital Solutions*. Validate the digital processes in a controlled environment.
7. *Scale and Optimize Digital Maintenance*.

3. CURRENT STATE OF MAINTENANCE MANAGEMENT ASSESSMENT.

Prior to digitization, the existing maintenance strategies and tools focused on the process of obtaining the CAS are evaluated. The machinery is subject to a maintenance plan made up of the set of tasks that are carried out on the machine in order to preserve the operational state of the vehicle and its technical characteristics which, in terms of safety, reliability, technical compatibility, health, environmental protection and, where appropriate, interoperability, were required in accordance with the provisions of the ETH (technical specifications for approval).

Maintenance plans focus on guaranteeing productive function and obtaining the aptitude for service, focused on the ability to circulate, which will be studied.

To measure the current state of the company we rely on the IMA dual maturity matrix: Maintenances Maturity vs Digital Maturity, Matrix. In accordance with the legal requirements imposed, the company is at a low level of Asset Management Maturity carrying out maintenance (Level 2: Developing/Bureaucratic: Formal systems in place but bureaucratic). The company has a Low Digital Maturity, with Basic digital record-keeping and reporting tools. Focusing on the study process, currently, the company carries out maintenance tasks with direct measurements from the operators and entered reports in paper status.

4. DEFINITION OF DIGITALIZATION GOALS AND OBJECTIVES.

As outlined in the Maintenance Maturity vs. Digital Maturity Matrix, the goal is to reach Level 4 Asset Management Maturity by leveraging advanced digital technologies, including digital twins and artificial intelligence (See Figure 2).

In this case study, the objective is to fully digitize the process of obtaining the CAS. Measurement variables currently recorded by operators will be entered digitally using software, eliminating the need for paper. These variables will be stored within the data infrastructure, enabling the digital twin to access and visually display them as needed.

GFMAM Asset Management Maturity Levels vs. Forrester R's Digital Maturity Levels.	**Low Digital Maturity (Basic Tools, Isolated Systems)**	**Developing Digital Maturity (Partial Integration, Early IoT)**	**Integrated Digital Maturity (Predictive, IoT Integration)**	**Advanced Digital Maturity (AI, Digital Twins, Full Integration)**	**Leading-Edge Digital Maturity (Industry Leadership, Continuous Innovation)**
Level 0: Innocent/Pathological (No recognition of asset management needs)	Basic spreadsheets or manual methods for asset management.	Minimal digital systems, mostly isolated without integration.	Early-stage CMMS with little or no real-time monitoring.	Limited use of digital monitoring, few integrations with existing platforms.	Basic AI experimentation but with little strategic focus.
Level 1: Aware/Reactive (Basic recognition but reactive management)	Initial efforts to implement digital solutions, possibly basic CMMS or EAM.	Partial IoT integration for asset monitoring; focus remains reactive.	CMMS linked to preventive maintenance; minimal analytics.	Predictive analytics starting to be adopted in select areas.	More advanced AI-driven processes in key areas, limited strategic alignment.
Level 2: Developing/Bureaucratic (Formal systems in place but bureaucratic)	Basic digital record-keeping and reporting tools.	Standardized tools like CMMS/EAMS but no optimization or real-time monitoring.	IoT and sensors deployed in critical assets, early analytics in APM.	Data-driven decision-making using AI and advanced analytics for key assets.	Fully integrated digital ecosystem leveraging AI for proactive decision-making.
Level 3: Competent/Proactive (Aligned with ISO standards, proactive)	Basic dashboards with manual data integration for asset tracking.	Partial integration of data through APM for PdM and condition monitoring.	Comprehensive IoT and predictive tools in APM connected to major assets in EAM.	Advanced AI and predictive analytics used for strategic decisions with basic AIP.	Continuous AI-driven optimization across the asset lifecycle with digital twins in APM/AIP.
Level 4: Optimizing/Generative (Optimized asset management aligned with full value)	Isolated advanced tools, limited by non-integrated systems.	Widespread adoption of IoT and basic AI but lacking real-time optimization.	Predictive models optimized with IoT, but not yet full digital twin integration.	Full digital twin implementation, AI-supported real-time optimization. Advanced APM-AIP.	Industry-leading proactive asset management based on real-time optimization and digital twins.
Level 5: Excellent/Leadership (Industry-leading practices, continuous improvement)	Minimal digital strategies beyond basic asset tracking.	Advanced analytics for monitoring asset health but lacking a full digital strategy.	Integrated AI systems for predictive and proactive asset management.	Comprehensive digital twin ecosystems; AI-driven asset management strategies.	Fully optimized, AI-powered AM framework EAM-APM-AIP driving continuous innovation and business value.

Figure 2. showing digitalization goals In the IMA dual matrix

5. SELECTION AND IMPLEMENTATION OF DIGITAL TECHNOLOGIES.

From a technological perspective, the objective is to create a digital twin integrated with real-time data streams to optimize asset performance throughout its lifecycle. Additionally, Enterprise Asset Management (EAM) and Asset Performance Management (APM) systems will be implemented to support predictive maintenance. However, the case study focuses on predetermined maintenance, which does not require the complexity of advanced predictive models.

For the digitalization of maintenance plans related to obtaining the Certification of Aptitude for Service (CAS), it is essential to develop an “Asset Identification Model”. This model will be applied to machinery components, workshop tools, company human resources, and the reports required for certification. In subsequent sections, the variables necessary for these data models are detailed.

Figure 3. Sensor, geolocation device with solar panel, RFID-NFC reader, NFC watch and processing unit

Although the IoT connection is not strictly required for obtaining the CAS, it plays a crucial role in the broader digitalization of rail transport. The IoT ecosystem facilitates the physical connection of the digital twin, enabling the tools necessary for digitizing workshop processes. Once deployed, the digital twin automates inspection processes by recognizing machines, equipment, technicians, and assigned resources. It manages permits and certificates, informs technicians of tasks, and guides them through a mobile interface. This structured data entry system generates alerts based on measurements, tracks task execution times, and ensures that all inspections are traceable and verifiable, thereby guaranteeing process quality.

The project's functional scope includes the conceptualization, design, and development of a comprehensive IoT ecosystem aimed at transforming railway maintenance. This initiative encompasses real-time data collection, advanced analysis, and automated decision-making to optimize the maintenance of the Spanish railway fleet. The IoT connection prototype is designed to create a robust and scalable infrastructure that ensures seamless integration with existing systems and provides a strong foundation for future expansion.

The IoT solution for the project is built on a distributed architecture that supports efficient real-time data collection, advanced data processing and analysis, and automated actions derived from insights. The solution comprises several critical components, detailed in Table 1 and in Figure 3.

Category	Components	Description
Data Capture	**IoT Sensor Nodes**	Data collector devices equipped with a variety of sensors to collect operational and environmental status data from trains and their surroundings. Essential for real-time monitoring.
		Geolocation devices with solar panels will provide precise train location data and temperature information, aiding fleet management and coordination.
	Automatic Identification Devices	Technologies using RFID and NFC for fast and accurate identification of railway components and personnel, facilitating inventory tracking. NFC Watched worn by maintenance personnel will identify individuals during measurement tasks and ensure precise tracking of intervention steps.
	Processing units	Components that preprocess sensor data, filtering and condensing it before transmitting to the central analysis platform, optimizing data transmission.
Data Processing and Analysis	**IoT Communication Gateways Network**	A dedicated network for transmitting data collected by sensor nodes to a centralized system. Operates across wide geographical ranges, ensuring uninterrupted and secure data coverage.
	Data Analytics Cloud Platform	Central system for processing and analyzing large volumes of data using advanced cloud solutions, AI, and ML algorithms. Provides predictive maintenance alerts and optimized recommendations.
Data Distribution and Visualization	**User Interface and Maintenance Management System**	Intuitive applications and dashboards for maintenance technicians and operators to access relevant information, customized maintenance plans, and tools for data-driven decision-making.
Data Storage	**Storage System**	Scalable and secure storage solutions designed to handle the exponential growth of data generated by the IoT system.
	Policy-Based Data Management	Policies regulating the lifecycle of data, from capture to archiving or disposal, ensuring compliance with privacy and data protection regulations.

Table 1. Solution critical components

6. CONSTRUCTION OF THE DATA INFRASTRUCTURE.

This section develops the data models and data architecture involved in study (according to the IMA Document requitements). The section includes also the platform designed in the project for the company to implement the digital twin.

6.1. Data models to obtain the CAS.

In this case study, the models to be developed for the digitization of the process are (Figure 4): data model of the machinery (asset), data model of the workshop (facilities) and data model of the variables involved in the inspection process (process) leading to the locomotive's Certificate of Aptitude for Service (CAS). These data models must be scalable. Currently, only the variables strictly necessary for obtaining the CAS are defined. However, additional variables will be identified and incorporated into the machine data model at a later stage. This scalability has been considered in the design of the digital twin.

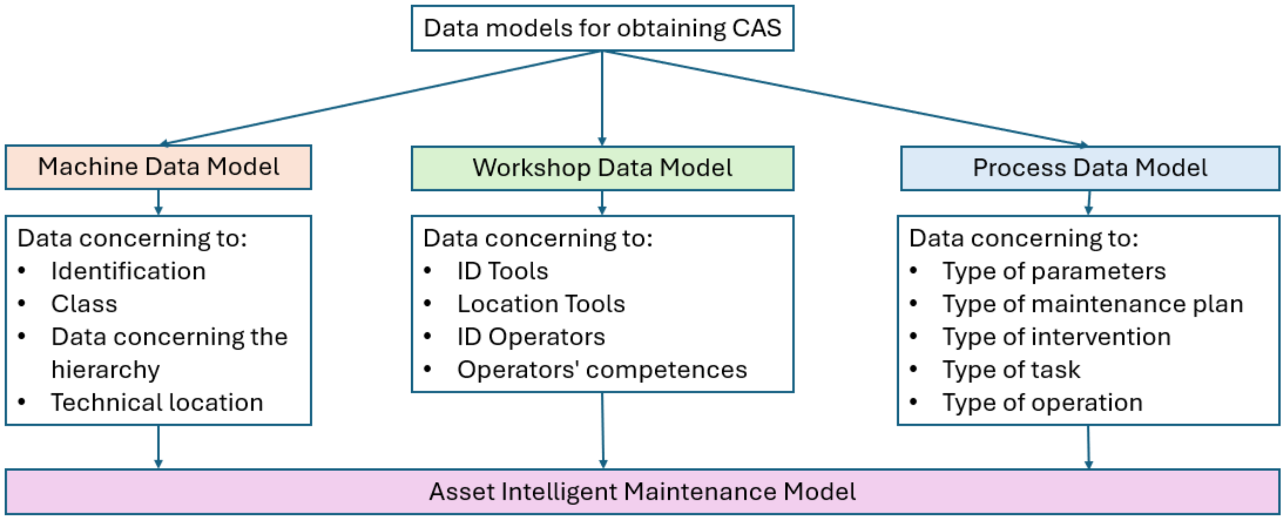

Figure 4. Data model for obtaining the CAS, which is the Asset Intelligent Maintenance Model.

The data models specify the variables essential for digitizing the CAS process. This digitization involved two key steps: first, defining the data models, and second, obtaining the ontologies that establish the relationships between these data models.

6.1.1. Asset Data Model – Machinery

The asset data model is defined below. There are four types of data to describe an asset: Asset Definition Model, Asset Criticality Model, Asset Monitoring Model and the Intelligent Asset Management Model.

Asset definition model:
For this data model, it is essential to implement a coding system for the various variables associated with the machinery. This coding enables consistent referencing throughout the asset's lifecycle. A hierarchical structure is used, breaking down the machinery into systems, subsystems, and components, thereby allowing variables to be linked to specific machine components. The alphanumeric code in Figure 5 illustrates how machine components are coded.

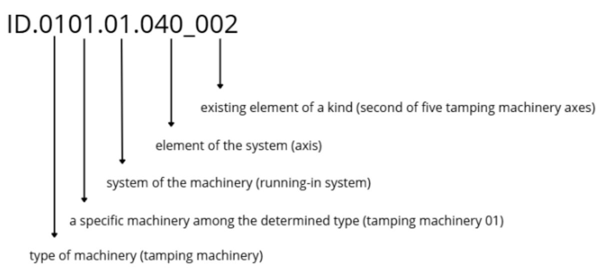

Figure 1. Identification code

Once the variable codification is established, the asset definition model is created. This model is further organized into four data subsets: Physical Asset Registration Data, Asset Functional Location, Asset Class Data, and Asset Reference System Data. Table 2 illustrates the variables used to identify different machines, using the tamping machine as an example. These variables form part of the tamping machine's data model and are critical for digitizing the CAS process, as they enable the identification of machinery and determination of the specific machine responsible for performing maintenance tasks.

Machinery			
Property	**Description**	**Example**	**Dimension**
ID_MACHINERY	This key represents the unique reference to the asset/machine register.	ID.0101.00.000_0001	Physical Asset Registration Data
C_CODNUMMAC	Indicates which specific tamping machine is being referred to.	01	
T_NAME	Name of the machinery.	TAMPING01	
T_DESCRIPTION	Description of the machinery.	Type Matisa B-66-U	
ID_TYPEMAC	Code that identifies the type of machinery.	01	Asset Class Data
C_ABBREVIATION	Abbreviation of the type of machinery code.	TM	

Table 2. Asset Definition Model for Machinery.

Each machine's data model includes the variables presented in the table, providing clear identification and classification of the machine by type and class. In the machinery data model, such as for the tamping machine, a hierarchical structure must be established. This involves coding the systems that constitute the machinery and the components within those systems. Additionally, each system and component must have an identification code, along with defined relationships to other components, ensuring the hierarchy and position of each element are explicitly recorded. The data model for the machinery also specifies the variables to be stored for each of the thirteen systems that compose the database, as detailed in the accompanying Table 3.

System			
Property	**Description**	**Example**	**Dimension**
ID_SYSTEM	This key represents the unique reference to the asset/machine register.	ID.0101.01.010_0001	Physical Asset Registration Data
T_NAME	Name of the system.	01.ROLLING	
T_DESCRIPTION	Description of the system.	Rolling system	
ID_TYPESYS	Key that associates a type of system with a system.	01 / 72	Asset Class Data
C_ ABBREVIATION	Abbreviation of the type of system code.	RO	

Table 3. Asset Definition Model for Systems.

The tamping machine consists of thirteen systems. These systems are in turn made up of different components. Figure 6 shows the systems of the machine.

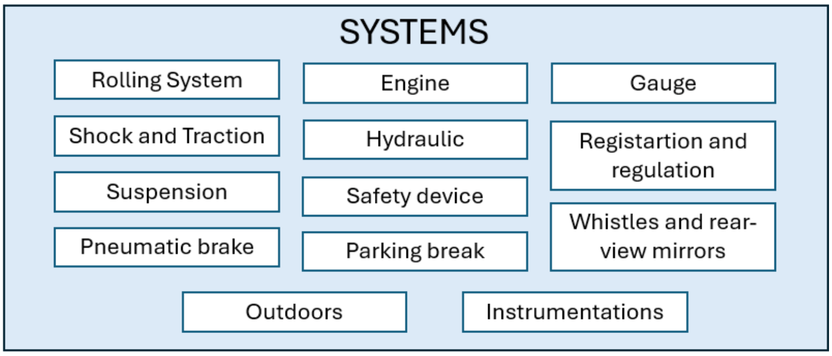

Figure 2. Systems of Tamping Machine.

The variables that are stored in the data model for each of the machine components are shown in Table 4. Here a code must be set for each component and the system to which the component belongs.

Components			
Property	**Description**	**Example**	**Dimension**
ID_COMPONENT	This key represents the unique reference to the component register.	ID.0101.01.020_0001	Physical Asset Registration Data
T_NAME	Name of the component.	BOGIE01	
ID_TYPECOMP	Reference to the type of component to which the component belongs.	020 / 16	Asset Class Data
C_ABBREVIATION	Abbreviation of the type of component code.	BG	

Table 4. Asset Definition Model for Components.

The asset definition model for the tamping machine includes the following variables: machine identification, definition and coordination of all systems, coding of components, and the functional location of each element. Functional locations represent the technical points that are part of the machinery, with each functional location record corresponding to a component within the hierarchical static structure of the asset/machinery. The diagram in Figure 7 illustrates the configuration of the rolling system, highlighting the hierarchy of elements. Properly defining this hierarchy is critical for digitizing activities. The accompanying image shows the hierarchy of elements and their respective locations within the tamping machine, a process repeated for all machine systems.

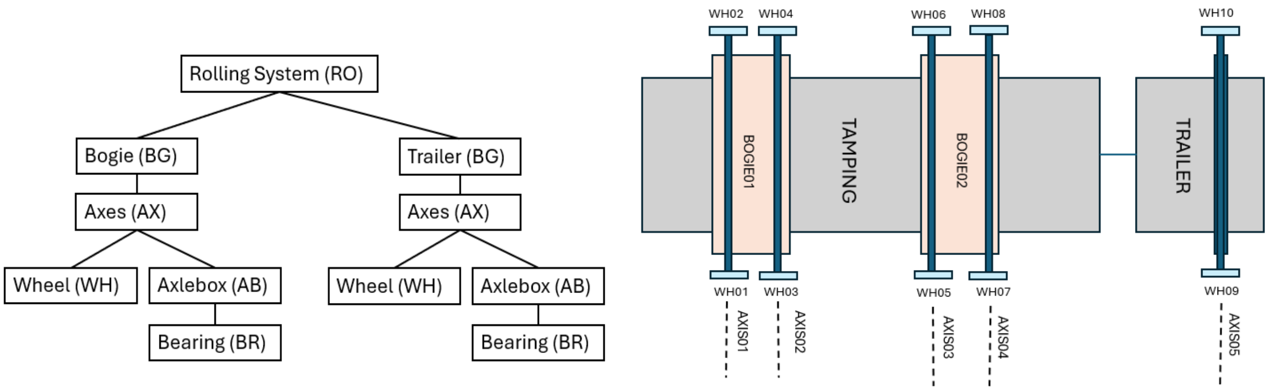

Figure 3. Rolling-in system schematic

Subsequently, we will define the functional locations within the machine of each of the systems and components. In the data model for each component the technical location of the component will be stored along with the variables shown in Table 5.

Technical Location			
Property	**Description**	**Example**	**Dimension**
ID_TECHLOCMAC	This key represents the unique reference to the record of the technical location.	FU.0101.01.050_0003	Asset Functional Location Data
ID_TECHLOCMAC_ORIGIN	Reference to the technical location of the item at the top level.	FU.0101.01.040_0002	
ID_MACHINERY	This key represents the unique reference to the asset/machine register	01	Physical Asset Registration Data
ID_SYSTEM	This key represents the unique reference to the asset/machine register.	ID.0101.01.010_0001	
ID_COMPONENT	This key represents the unique reference to the component register.	ID.0101.01.050_0003	
T_NAME	The name given to the technical location.	WHEEL03	
N_ORDER	A numeric field that indicates the order of the technical locations of the same type of component.	0003	
T_DESCRIPTION	Description of the technical location.	Left wheel of axis 02 of bogie 01	

Table 5. Asset Definition Model for Technical Location.

The data model for the machine involved in CAS acquisition tasks has been established, encompassing the identification of components, systems, and the machine itself, along with their functional locations. Four key tables were required: one for machine identification and classification, another for system identification and typology, a third for component identification and typology, and a final table for the functional locations of all elements. Each functional location is directly linked to the corresponding identified element, for a comprehensive representation of the machine's data.

6.1.2. Asset Data Model – Workshop

After defining the machinery's data model, the next step is to establish the workshop data model. This model should include the tools required to carry out the CAS checks, along with their functional locations. Additionally, it should account for the workshop's human resources, identifying personnel to track which operators are responsible for specific tasks.

Table 6 outlines the variables necessary to describe the tools in the workshop data model. For measurement devices, it is necessary to determine which type they are, since there may be more than one device of the same type (See Table 7).

Measuring Device			
Property	**Description**	**Example**	**Dimension**
ID_MEASDEV	Primary key that identifies the record in the DISMED entity. This key represents the unique reference of a measuring device.	ID.HER.0101.006_0001	Physical Asset Registration Data
T_NAME	Identifying name of the measuring device.	Caliber rolling parameter (22626)	
ID_TYPEMEASDEV	Reference to the device type.	006	Asset Class Data
ID_TECHLOCMEASDEV	Reference to the technical location of the tool used.	FU.HER.0101.006_0001	Asset Functional Location Data

Table 6. Asset Definition Model for Measuring Device.

Type of Measuring Device			
Property	**Description**	**Example**	**Dimension**
ID_ TYPEMEASDEV	Primary key that identifies the record in the TYPEMEASDEV entity. This key represents the unique reference of a type of measuring device.	006	Asset Class Data
T_ TYPEMEASDEV	Identifying name of the measuring device.	Caliber rolling parameter (22626)	Physical Asset Registration Data

Table 7. Asset Class Model for Measuring Device.

Then, the variables of the workshop data model used to describe human resources are defined in Table 8.

Users			
Property	**Description**	**Example**	**Dimension**
ID_OPERATOR	Primary key that identifies the record in the USERS entity. This key represents the unique reference to a user's record.	ID.RHH.0000.001_0001	Physical Asset Registration Data
T_NAME	The user's name.	José Santos	
T_SURNAME	The user's surname.	García Valencia	

Table 8. Definition Model for Human Resources.

6.1.3. Inspection Process Data Model.

The data model for the CAS inspection process has been developed, encompassing all parameter types necessary to define CAS tasks. It specifies the types of maintenance plans that are performed, as well as the interventions included within those plans. Each intervention details the types of tasks it comprises, and each task, in turn, describes the types of operations it includes. Additionally, the CAS assigns a value to each operation, so the model also incorporates the types of parameters that can be associated with each operation (See Figure 8).

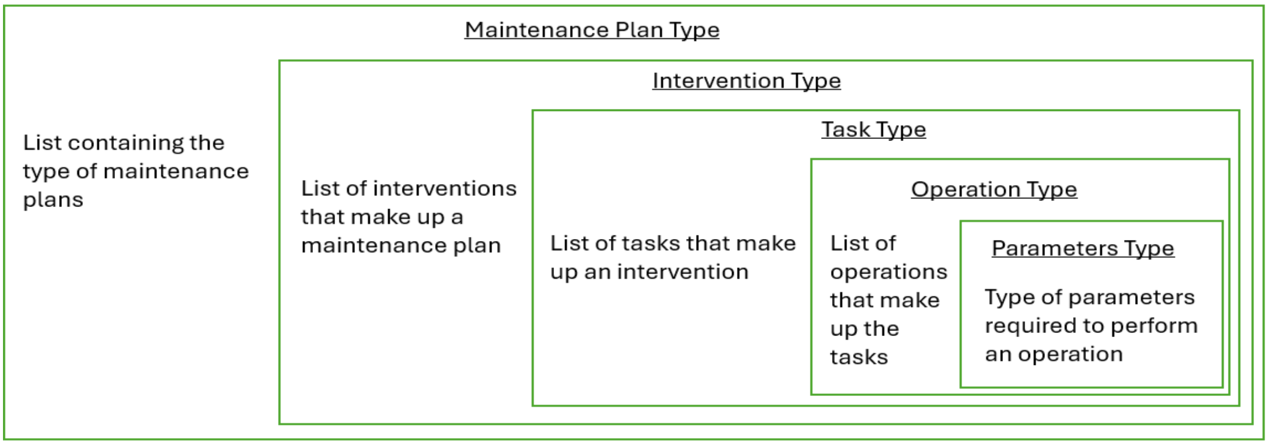

Figure 8. Type of data for the CAS.

First, the types of maintenance plans applicable to different machines are outlined (Table 9). The variables required for the data model to describe these maintenance plans include the plan's identification code and the reference to the machine on which it is performed.

Type of Maintenance Plan		
Property	**Description**	**Example**
ID_TYPEMANPLAN	This key represents the unique reference to a type of maintenance plan.	1
C_TYPEMANPLAN	Alphanumeric code that uniquely represents the type of maintenance plan.	AZVI1416
T_TYPEMANPLAN	Description of the type of maintenance plan.	Tamping machine maintenance plan
ID_TYPEMAC	Type of machinery to which it refers.	01

Table 9. Data Model of type of Maintenance Plan.

The various interventions required to obtain the CAS are outlined In Table 10. These interventions are part of the maintenance plans. The table below displays the codes used to identify the different types of interventions.

Type of Intervention		
Property	**Description**	**Example**
ID_TYPEINT	This key represents the univocal reference to a type of intervention.	4 5 6
C_TYPEINT	Alphanumeric code that uniquely represents the type of intervention. IS1: Every year. IS2: Each two years. IM: Each ten years.	IS1 IS2 IM
T_ TYPEINT	Name of the intervention.	IS1 IS2 IM
N_DURATION	Length of time the intervention lasts.	1 2 10

Table 10. Data Model of type of Intervention.

Table 11 presents the various tasks that can be carried out as part of the interventions, which, in turn, involve different operations. Each type of task is assigned a code for easy reference.

Type of Task		
Property	**Description**	**Example**
ID_TYPETASK	This key represents the unique reference to a type of task.	554
C_TYPETASK	Numerical code that uniquely represents the type of task.	01.01
T_TYPETASK	Description of the type of task.	Wheel parameters
ID_TYPEINT	Reference to the type of intervention in which the task type is executed.	6
ID_TYPESYSTEM	Reference to the type of system that applies the task.	72

Table 1. Data Model of type of Task.

Tasks, in turn, consist of various types of operations. The data stored in the process data model to reference the types of operations performed are detailed in Table 12. Operations assign attributes to elements, and the data model defines constraints these attributes must meet, such as a lower boundary limit for measurements.

Type of Operation		
Property	**Description**	**Example**
ID_TYPEOPE	Primary key that identifies the record in the TIPOPE entity. This key represents the unique reference to a type of operation.	01.01-1 01.01-2 01.01-3
C_TYPEOPE	A numeric code that represents the order of the type of operation within the type of task.	1 2 3
T_TYPEOPE	Name of the type of operation.	HEIGHT THICKNESS QR
ID_TYPEPAR	Reference to the type of parameter that the operation type has.	416 417 418
ID_TYPETASK	Reference to the type of task that the operation is part of.	554 554 554
ID_TYPEMEASDEV	Reference to the type of measuring device to which it refers.	ID.HER.0101.006_0001 ID.HER.0101.006_0001 ID.HER.0101.006_0001

Table 2. Data Model of type of Operation.

Each operation is assigned a specific variable value, as detailed in Table 13.

Type of Parameters		
Property	**Description**	**Example**
ID_TYPEPAR	This key represents the unique reference to the parameter type.	416
C_TYPEPAR	Code that uniquely represents the parameter type.	HEI
T_TYPEPAR	Description of the type of parameter.	HEIGHT
C_TYPEDATVALPAR	Code that represents the data type of the parameter value.	RANGE
N_LOWRANGE	Lower tolerance range. This field will be populated when the parameter type has a lower limit beyond which the measurement is out of tolerance.	/
N_UPPERRANGE	Higher tolerance range. This field will be populated when the parameter type has an upper limit beyond which the measurement is out of tolerance.	35
C_UNIMEAS	Code corresponding to the unit of measure of the parameter type.	mm

Table 3. Data Model of type of Parameters.

Defining operator qualifications for specific tasks is crucial, as not all operators can work on every system. Variables linking systems to qualified operators are recorded (see Table 14). The process data model defines all variables for the CAS, allowing interventions to be selected, operations executed, and attributes measured to ensure they meet the required criteria.

Type of System-User		
Property	**Description**	**Example**
ID_TYPESYSUSER	Primary key that identifies the record in the TYPESYSUSER entity. This key represents the unique reference to the registry of a system type that can be reviewed by a user.	01
ID_USER	Reference to the operator who can review the type of system referenced.	ID.RHH.0000.001_0001
ID_TYPESYSTEM	Reference to the system that can be reviewed by the referenced operator.	01

Table 4. Data Model of type of System and User.

6.1.4. The Intelligent Asset Management Model (the CAS)

The asset, workshop, and process data models have been defined for the CAS acquisition process. The Intelligent Asset Management Model records past and ongoing interventions, ensuring parameter compliance and operator qualification while storing all data for future reference. Operators use a digital device to select the maintenance plan, identify themselves, perform tasks, and input measurements. Tables 15 and 16 outline the data model, starting with assigning a unique code to the maintenance plan and detailing the intervention.

Maintenance Plan		
Property	**Description**	**Example**
ID_MAINPLAN	This key represents the unambiguous reference to the maintenance plan.	MM.0001.000.0000.00.00
ID_TYPEMAINPLAN	Reference to the type of maintenance plan.	1
ID_MACHINERY	Reference to the machine on which the maintenance plan is applied.	ID.0101.00.000_0001
D_START	Start date of the maintenance plan.	01/01/2023
D_END	End date of the maintenance plan.	31/12/2024

Table 5. Data Model of Maintenance Plan.

Intervention		
Property	**Description**	**Example**
ID_INTERVENTION	Primary key that identifies the record in the entity INTERVENTION. This key represents the univocal reference to an intervention.	MM.0001.001.0000.00
ID_TYPEINT	Reference to the type of intervention.	6
ID_MAINPLAN	Reference to the maintenance plan in which the intervention is planned.	MM.0001.000.0000.00
D_START	Date of start of the intervention.	01/01/2023
D_END	Date of end of the intervention.	01/02/2023
C_RESULT	Code with the result of the intervention.	Positive Negative Corrected

Table 6. Data Model of Intervention.

The data model contains a list of all tasks performed during the intervention, with each task identified by a code that links it to a specific intervention task type (see Table 17), as defined in the process data model.

Task		
Property	**Description**	**Example**
ID_TASK	Primary key that identifies the record in the TASK entity. This key represents the univocal reference to the task.	MM.0001.001.0101.00
ID_TYPETASK	Reference to the type of task.	534
ID_INTERVENTION	Reference to the intervention to which the task belongs.	MM.0001.001.0000.00
D_START	Date of start of the task.	01/01/2023
D_END	Date of end of the task.	15/01/2023

Table 7. Data Model of Task.

The list of operations performed includes each operation identified by its type (as in Table 18), as defined in the process data model. Multiple operations of the same type may occur within the same task of an intervention and maintenance plan, but they must always take place in distinct time periods. This ensures flexibility for scenarios where a measurement operation yields a negative result, necessitating repeated attempts.

Operation		
Property	**Description**	**Example**
ID_OPERATION	This key represents the unique reference to the operation.	MM.0001.001.0101.01
ID_TYPEOPE	Reference to the type of operation.	01.01-1
ID_TASK	Reference to the task of which the operation is part.	MM.0001.001.0101.00
D_START	Date of start of the operation.	01/01/2023
D_END	Date of end of the operation	05/01/2023
C_RESULT	Result of the operation that can take the values of a list called *result*.	Successful Unsuccessful

Table 8. Data Model of Operations.

In the CAS acquisition process, it is essential to record the location where each measurement is performed, whether it is in the workshop or another space. To achieve this, the variables in Table 19 must be utilized.

Localization		
Property	**Description**	**Example**
ID_LOCALIZATION	Primary key that identifies the record in the LOCALIZATION entity. This key represents the unique reference to a specific location.	001
ID_GEOLOCALIZATION	Code of the geolocation to which it refers.	001*
C_LOCALIZATION	Code alphanumeric that represents univocally the localization.	Workshop
T_LOCALIZATION	Description of the localization.	Sevilla workshop

* This table associates a location with its corresponding geolocation. The coordinates of each location are stored in the Geolocation Table (Table 10).

Table 9. Data Model of Measurement Localization.

This entity contains a comprehensive list of all measurements performed, encompassing information such as the type of operation within the associated maintenance plan, the responsible operators, and the measuring devices used. The data model is employed to validate the accuracy of the measurements performed by the operator. For instance, it verifies whether the measured value (T_VALUE) falls within the permissible limits for the specified parameter (see Table 20).
Once all the data models have been developed, the next step is to create the ontologies that define the relationships between them. The schema below illustrates these connections. For example, measurements are linked to the operator who performed them, the associated parameter type, the tool used, and the specific machinery or component being measured.

Measurement		
Property	**Description**	**Example**
ID_MEASUREMENT	This key represents the unambiguous reference to measurement.	MM.0001.001.0101.01.01
ID_OPERATION	Reference to the operation, related to a certain measurement, that is part of the task associated with a certain intervention within a maintenance plan.	MM.0001.001.0101.01.00
ID_TYPEPAR	Reference to the type of parameter measured.	416*
ID_MACHINERY	Reference to the machinery from which the measurement is taken.	ID.0101.00.000_0001
ID_COMPONENT	Reference to the measured component.	ID.0101.01.050_0003
ID_USER	Reference to the user who takes the measurement or supervises it.	ID.RHH.0000.001_0001
ID_MEASDEV	Reference to the type of measuring device used to perform the measurement.	ID.HER.0101.006_0001
ID_LOCALIZATION	Reference to the location where the measurement is taken.	001

Measurement		
Property	**Description**	**Example**
T_VALUE	Value of measurement. The field type is given by what is indicated in the parameter type .	29,5
B_PROCESSED	An attribute that indicates whether the measurement has already been processed.	True False
B_CHECK_VISUAL_CORRECT	The visual check field is used so that if in a visual check it's seen that the component clearly needs to be replaced, a measurement that may introduce noise does not have to be entered.	"S": Correct visual check. "N": Check not correct. "NA": No applies.

*Code that refers to the parameter to be measured in this operation. (Table 3)

Table 20. Data Model of Measurement.

Similarly, operations are associated with their corresponding area, which is connected to the intervention, and the intervention is linked to the overarching maintenance plan (See Figure 9).

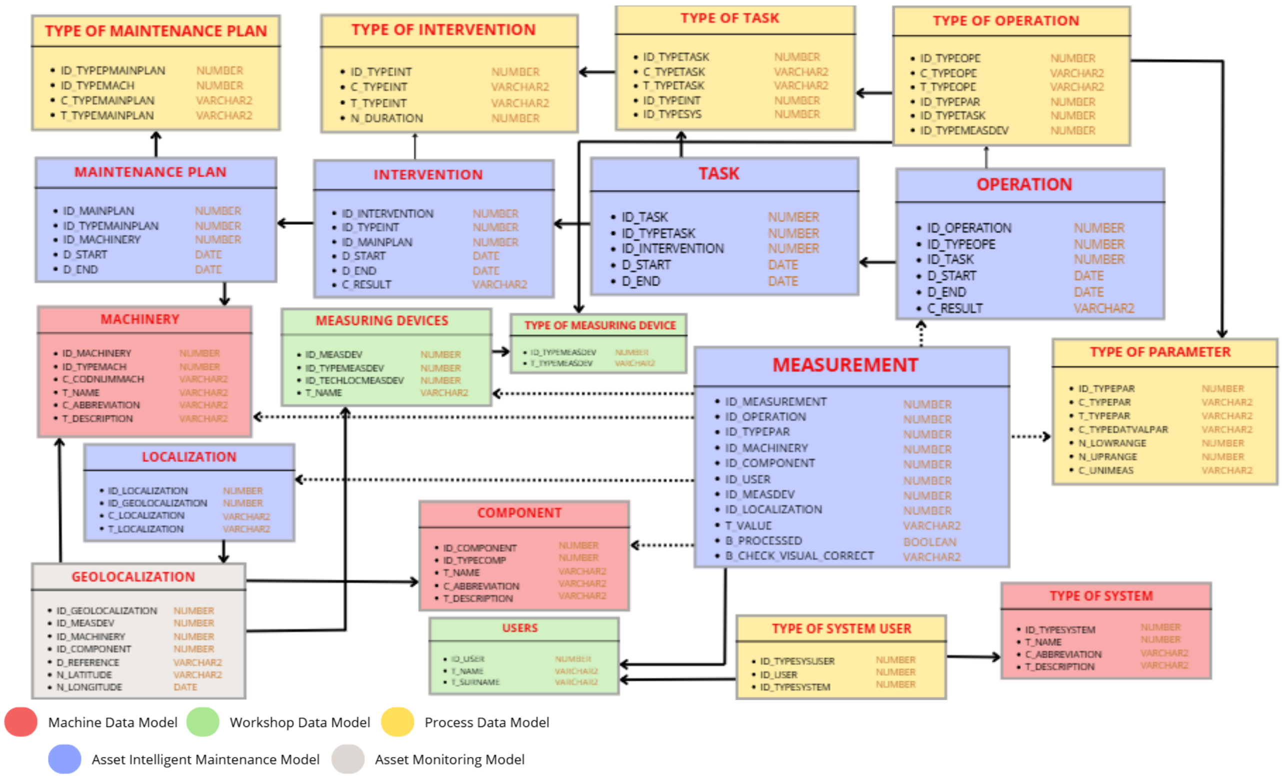

Figure9. Ontologies connecting data models.

The asset definition model and the intelligent asset management model have been established for the digitalization of the CAS acquisition process. To provide a comprehensive example of the IMA document data models, an asset monitoring model and an asset criticality model will also be developed. While these models are not essential for describing the CAS process, they are included to illustrate a complete example of asset digitization.

6.1.5. Asset Monitoring Model

As an example, a data model will be developed to digitize information from a solar-powered GPS installed on one of the machines. The purpose of using geolocation devices with solar panels is to provide each railway vehicle with an advanced geolocation system capable of transmitting real-time data on the train's location and operational status. Solar-powered devices ensure energy self-sufficiency, significantly reducing maintenance requirements. This data model (in Table 21) connects the solar GPS to the machine it is installed on, specifying its location and any related components. It also links the geolocation data to the measuring device (the GPS unit), recording essential attributes such as latitude, longitude, and the timestamp of each measurement.

Geolocation		
Property	**Description**	**Example**
ID_GEOLOCALIZATION	Primary key that identifies the record in the GEOLOCALIZATIONS entity. This key represents the unique reference of a GPS position in which the machine is located.	015
ID_MEASDEV	Reference to the measuring device.	ID.HER.0101.024_0001
ID_MACHINERY	Reference to the machine.	ID.0101.00.000_0001
ID_COMPONENT	Id of the component it references. It can be null if the geolocation refers to the complete machine that will be most of the records.	ID.0101.01.050_0003
D_REFERENCE	Reference date of when the GPS position is taken.	01/01/2023
N_LATITTUDE	The latitude at which the component or machine is positioned.	37.3751
N_LONGITUDE	The length at which the component or machine is positioned.	57.3751

Table 10. Data model of Geolocation.

6.1.6. Asset Criticality Model

This section presents an example of criticality analysis, a fundamental step in maintenance management for complex engineering systems such as railway infrastructure and rolling stock. This analysis is a prerequisite for assessing and optimizing existing maintenance programs. The primary goal of the Asset Criticality Model is to prioritize assets based on their criticality. Critical assets are those whose failure would have a significant impact on overall system performance, safety, or regulatory compliance. By evaluating criticality, organizations can focus their resources and attention on assets that have the greatest influence on operational safety, performance, and continuity.

Using a multi-criteria analysis approach, this example focuses on the tamping machine. The model consolidates various factors affecting equipment criticality into a single score, reflecting its criticality level. This scoring system is instrumental in decision-making, ensuring that the most essential assets receive appropriate resources and attention. Implementing this process reduces the risk of unexpected failures while enhancing the efficiency and resilience of railway infrastructure. To achieve this, a structured procedure is followed to identify the necessary factors, assign scores, and determine the methodology for combining them.

Criticality is calculated as the product of two factors: **severity** and **frequency** of a functional loss.

- **Severity** represents the consequences of an asset's loss of function, evaluated across four dimensions: safety, availability, quality of service, and the cost of corrective maintenance (CM). To each dimension the organization assigns a weight contributing to the overall severity value: safety (10%), operational reliability (30%), quality of service (20%), and CM cost (40%).
- **Frequency** measures the likelihood of failure, categorized into four levels based on the number of annual failures.

Table 22 outlines the frequency factors used to determine criticality based on failure rates (considered here as failures per million kilometers – FPMK).

Annual frequency of failures FPMK	**Ranking**	**Frequency factor**
$f \geq 2$	Very High	2
$1 \leq f < 2$	High	1.5
$0.5 \leq f < 1$	Medium	1.2
$f < 0.5$	Low	1

Table 11. Frequency factors levels.

To calculate the severity of an asset for each factor, four levels are defined. For safety, the MIL-STD-882C standard is applied. For reliability, quality of service, and CM cost, the levels are determined based on the potential costs incurred from the functional loss of the asset. The following table outlines the relative values for the various effects across each criterion. For example, a functional loss impacting quality of service—such as one affecting the entire machinery—may result in a cost of $4,000, as shown in Table 23. Similarly, an operational reliability issue that causes a system stoppage exceeding a specific threshold (e.g., more than *x* minutes) might incur a cost of $10,000.

Within the safety and reliability factors, there exists a category designated "non-admissible", indicating catastrophic failure consequences. Any asset falling into this category is automatically classified as critical, regardless of the weight assigned to other factors.

Criteria to measure Severity (e.g. based on penalization cost and CM budget, $)			
Safety criteria (weight:10%)	Operational reliability (weight:30%)	Quality service (weight:20%)	CM Cost (weight:40%)
Non admissible	Non admissible	4000	400
Critical	10000	2000	200
Marginal	5000	800	60
Negligible	0	0	10

Table 12.Effects per functional loss.

For the above levels, the severity weighting weights must be determined. The level with the highest consequence will have a weight equal to the weight of the criterion in which it is found, unless it is an unacceptable consequence, which will have a weight equal to 100. Table 24 presents the corresponding severity values of the effects for a given functional loss. For example, a functional loss of quality service resulting in a cost of more than $4000 will count for 20 severity points (up to 100). Note how an unacceptable effect of a functional loss will count for 100 (maximum value) regardless of the effect on any other criteria.

Criteria to measure Severity			
Safety criteria (weight:10%)	Cost related criteria (e.g. based on penalization cost and CM budget, $)		
	Operational reliability (weight:30%)	Quality service (weight:20%)	CM Cost (weight:40%)
Category of effects per criteria and functional loss			
100	100	20	40
10	30	10	20
6.6	15	4	6
0	0	0	1

Table 13. Frequency and Potential quantitative effects.

Table 25 shows the complete data values of the frequency and severity categories, assigned for each asset, considering the weights of each criterion and the logic explicated in the previous paragraph. SC columns refer to the severity criteria considered for the analysis carried out. (SC1: Safety; SC2: Operational Reliability; SC3: Quality Service; SC4: CM Cost).

Assets (systems)	Frequency	Frequency factor	SC1	SC2	SC3	SC4	Severity	Criticalit y
Rolling	0.2	1	100	100	10	20	100	100
Shock and Traction	0.1	2	100	30	10	20	100	100
Suspension	1.5	1.5	10	30	4	20	64	96
Pneumatic brake	0.3	1	100	15	4	6	100	100
Parking Brake	7	2	6.6	15	10	20	51.6	100
Engine	0.8	1.2	6.6	100	20	40	100	100

Table 14. Frequency and Potential quantitative effects per different assets.

Table 26 defines the data model for each asset, capturing the criticality aspects.

Entity	Attribute	Description	Example
Asset	ID Asset	Unique identifier for the asset.	ID.0101.01.010_0001
	Functional Location	Functional location of the asset.	FU.0101.01.010_0001
	Severity	Severity level associated with the asset.	100
	Criticality	Criticality level of the asset.	100
Frequency	ID Frequency	Unique identifier for the frequency factor.	FQ.0101.01.010_0001
	Frequency	Annual frequency of failures.	0.2
	Frequency Factor	Weight associated with the frequency.	1

Entity	Attribute	Description	Example
Safety	ID Safety	Unique identifier for the safety factor.	SF.0101.01.010_0001
	Safety	Value according to MIL-STD-882C standard	Catastrophic
	Safety Factor	Weight associated with the safety factor.	100
Operational Reliability	ID Operational Reliability	Unique identifier for the operational reliability criteria.	OR.0101.01.010_0001
	Operational Reliability Cost	Value according to estimated cost values in dollars.	20 000
	Reliability Factor	Weight associated with the operational reliability criteria.	100
Quality Service	ID Safety	Unique identifier for the quality service criteria.	QS.0101.01.010_0001
	Quality Service Cost	Value according to estimated cost values in dollars.	3 000
	Quality Factor	Weight associated with the safety factor.	10
CM Cost	ID CM Cost	Unique identifier for the CM cost criteria.	CM.0101.01.010_0001
	CM Cost	Value according to estimated cost values in dollars.	300
	CM Factor	Weight associated with the CM cost criteria.	20

Table 15. The asset criticality model.

6.2. The necessary architecture for the CAS process digital twin.

The development of the data architecture and the design of the technological platform for the digital twin have been carried out with the collaboration of specialized companies. Their expertise has contributed significantly to ensuring the robustness, scalability, and functionality of the solution. Below is the solution that the company should adopt to define the architecture of the digital platform for a digital twin. The data infrastructure is divided into layers, each serving distinct functions.

The goal is to create a solid, modern, modular, extensible, and scalable architecture that meets the current and future requirements of digital twins. The criteria used for selecting the various components are as follows:

- Compliance with standards.
- Use of consolidated components that are widely used in the market, primarily based on Apache Software Foundation projects.
- Adoption of open-source projects with no associated licensing costs.
- Ease of integration of new components/technologies.

- Ease of maintenance and sustainability of the complete solution.
- 100% on-premises deployment: the chosen solution must allow deployment and management entirely within proprietary hardware infrastructure.

The layers described below represent different levels of abstraction and functionality within a data and technology architecture that was selected (see Figure 10).

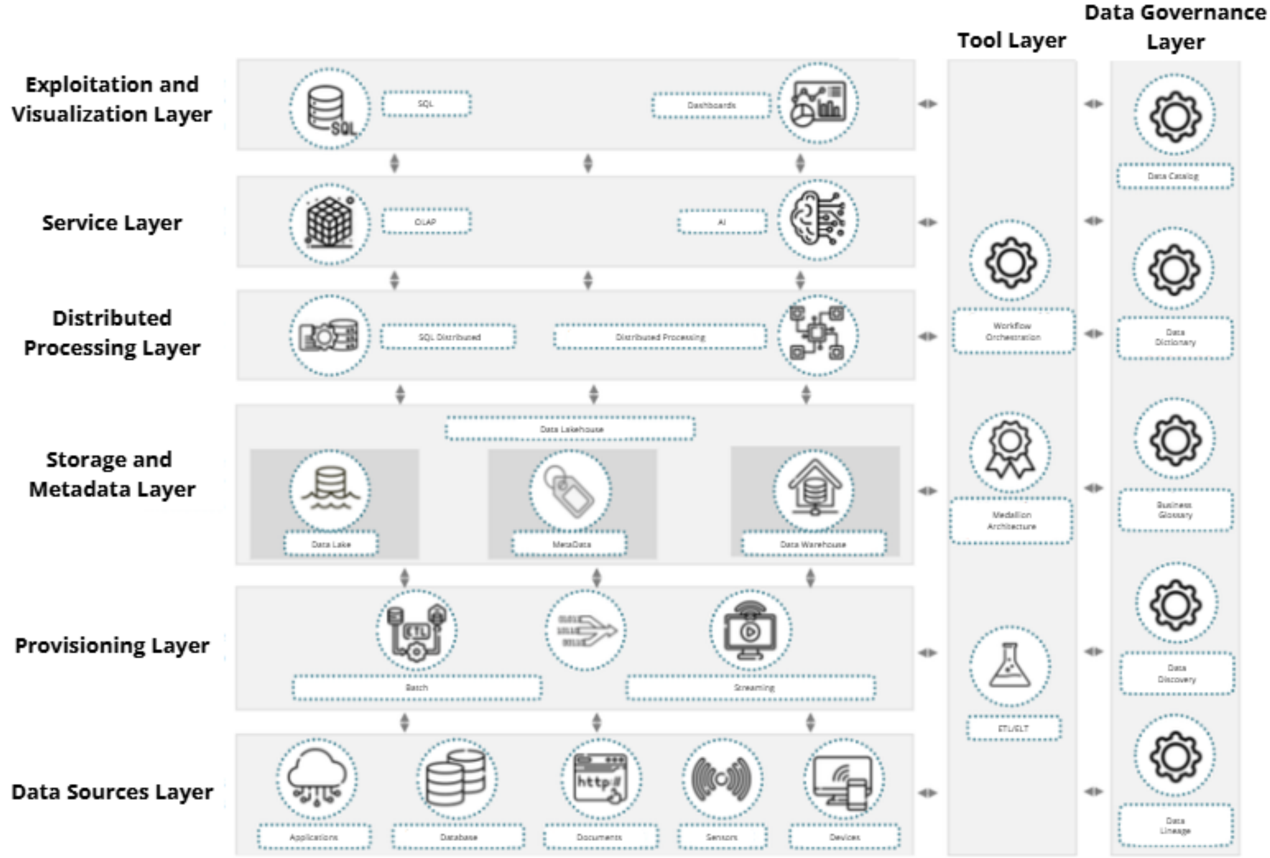

Figure 10. Conceptual schematic of the architecture of the digital twin.

- *Data sources layer:* It is not strictly an element of the architecture, but it is necessary to have this layer as it represents the possible data sources_(sensors, CMMS application, cloud applications, databases, devices, web pages, etc.). Data is heterogeneous (structured, semi-structured, or unstructured data).
- *Provisioning layer:* Layer responsible for connecting to the different data sources described in the previous layer to be able to carry out the processes of: Data extraction and Hosting of data in the storage layer. This layer must be able to process data in both real-time streaming and batch data streams. It is the solution for provisioning information and ingesting data into the Lakehouse, covering: the historical data load and the data collection phase (regardless of whether the data is structured, semi-structured, or unstructured). After identifying the data sources, it is necessary to design the data collection mechanisms that will enable data import.
- *Storage and Metadata Layer:* This layer is responsible for hosting and organizing all the data that is published on the platform (Data Space). It must allow horizontal scaling and must perform a distributed storage in a transparent way for the rest of the platform layers. It must facilitate easy access to data, providing a semantic and organizational layer while providing services and data with aspects such as security and information quality. This layer is typical

of a *Data Lakehouse* and includes all metadata for tables and complete *datasets* (including information about the source, structure, relationships and usage patterns of the data).

- *Processing layer:* This layer is responsible for executing all the processes of data cleansing, extraction, transformation, aggregation, training of *Machine Learning models, etc*. As in the storage layer, the processing layer must allow horizontal scaling to increase processing capacity when necessary. This layer must have the capacity to work with data stored in a distributed manner (distributed among the different servers that make up the processing cluster).
- *Services layer:* Layer responsible for providing value-added functionalities to the platform, such as:
  - Advanced Analytics Frameworks (machine learning, AI).
  - Enable access to distributed storage layer as database services (SQL).
  - Multidimensional OLAP models that enable descriptive analytics.

  This layer is intended to take full advantage of the compute and distributed storage layers.
- *Data exploitation, visualization and publication layer:* The platform will have a large amount of data of enormous quality that is of interest. This layer is responsible for carrying out the necessary steps to publish this data securely (Dashboard).
- *Consumption layer:* This layer is responsible for getting the most out of the storage and processing layers by providing value-added functionalities to the platform, including digital twin simulations, the use of *advanced analytics frameworks (*Business Intelligence*),* interactive searches, dashboards or consumption of trained predictive models.
- *Data governance layer:* Data governance refers to the collection of practices, policies, and roles related to the acquisition, management, and effective utilization of data. Data governance confirms the quality and security of an organization's data, determining who can use what data and when. Among the functionalities offered by this layer, we can find:
  - Data discovery services that enable reuse.
  - Automated detection of sensitive data (PII).
  - Data Quality Services.
  - Data lineage capacity: having a record of where the data originated, as well as the movements and transformations it has undergone over time.
- *Tool layer:* It is the layer that is responsible for providing end users with a "toolbox" that allows them to carry out their work in a comfortable, efficient and reusable way. Among this toolbox we find functionalities as diverse as the following:
  - Tools for data management through the Medallion *Architecture*, to assist in transformations of source data, gaining new insights.
  - Notebook tools for the creation and testing of *Machine Learning* models, which will later be deployed through the Service Layer.

6.3. Technological solution adopted for the digital twin architecture.

Once the appropriate architecture was established, the selection of technological solutions for each layer was completed. A detailed layer-by-layer analysis identifies the specific software platforms assigned to perform the functions of each layer, ensuring seamless integration across the system.

- *Provisioning layer*

For provisioning layer Lambda Architecture is used. The Lambda Architecture is a data ingestion configuration that consists of both real-time and batch methods. The configuration comprises batch, service, and speed layers. The first two layers index data in batches, while the speed layer instantly indexes data that has not yet been processed by the slower batch and service layers. This continuous transfer between the layers ensures that data is available for querying with low latency. Additionally, tools are provided for the execution and implementation of ETL and ELT processes.

ELT (Extract, Load, and Transform): This process is used in the most modern Data Warehouses. It begins by extracting data from source systems and depositing it into a Data Lake or Lakehouse with minimal transformation. The transformation step is carried out directly in the destination systems using integrated processing capabilities, without cleaning the data beforehand. This approach leverages the computational power of the data warehouse to transform the data as needed during the analysis phase. The ELT pattern emerged as a more suitable approach for processing data in Data Lakes and Lakehouses, characterized by:

- Extraction from sources: Data is extracted from the source to be stored in the Lakehouse.
- Load before transforming: In the Lakehouse, all extracted data is stored directly in a storage system like Amazon S3, Azure Blob Store, or Google Cloud Storage, in its original structure. Once stored, any transformations are performed directly in the Lakehouse on demand as different analytical tasks arise. The idea is to postpone the transformation processing cost until the analyses that require it are necessary. Since there is no predefined schema to restrict transformations in data lakes after loading, ELT provides greater flexibility for performing analytical tasks. This makes Lakehouses more adaptable and scalable to handle diverse types of data and large volumes of data. Instead of a monolithic ETL application, transformations are typically performed by a much less costly computation engine.
- Data exploration and discovery: Since raw data is preserved and various computing capabilities can be leveraged, Data Lakes and Lakehouses are excellent environments for data exploration, machine learning, and advanced analytics where it is necessary to discover unknown patterns from raw data.

APACHE KAFKA

kafka

The Kafka tool is used to perform real-time provisioning (Figure 11). Kafka guarantees that any data “flowing” through the Data Bus would arrive, in its raw (unprocessed) form, to the Data Lake. The idea behind this approach is to ensure that all data produced in the project environment is not lost and is stored in the Data Lake, regardless of whether or not such data will be used within the project (the Data Lake philosophy is precisely to offer the necessary storage and provisioning capabilities so that no data from the organization, in this case, from the project, has to be discarded, even if initially no use is seen for such data).

The following is an explanation of how procurement works using the Kafka tool.

- *Producers:* Producers are responsible for publishing data records to Kafka topics.
- *Topics:* Topics are categories (feeds) in which producers publish data records.

- *Partitions:* Each topic is divided into multiple partitions. Partitions allow horizontal scalability and parallel processing within a Kafka cluster. They increase performance.
- *Brokers:* Brokers are Kafka servers in the cluster that manage the storage and replication of data. Each broker manages one or more partitions and communicates with producers and consumers.
- *Consumers:* Consumers are applications or systems that subscribe to Kafka topics and consume the data records.

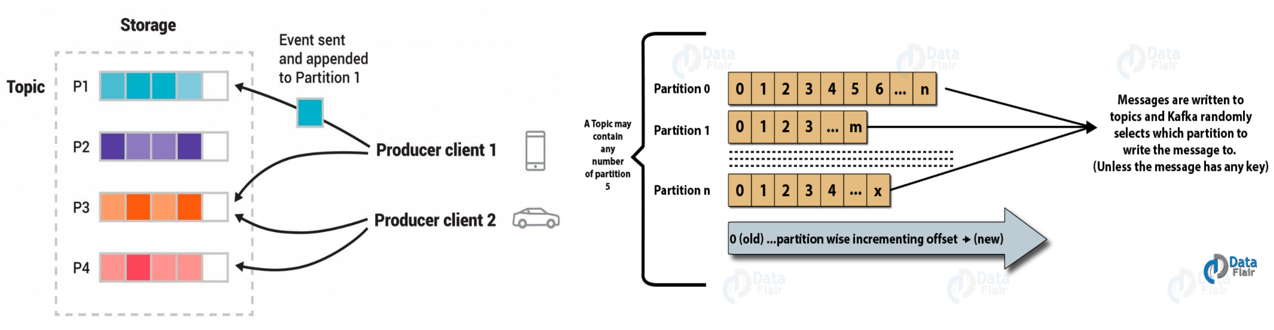

Figure 14. Operating scheme of Kafka.

- *Storage and Metadata Layer*

The following describes how the storage layer in which the data models used in the digitization of the case study are housed should be developed.

In the first instance, the data will be stored in a raw data format, as obtained from the different sources without cleaning. From here, three layers are established, depending on the level of processing and curation of the data (medal system):

The Bronze layer (Raw data). This layer comprises raw or raw data ingested from numerous data sources and preserves it as is.

- The table structure of this layer is like the table structure of the source system, with additional metadata columns containing: the upload date/time, file name, upload ID, and so on. One of the aspects to consider for maintaining storage in a Data Lake is to build a metadata catalogue that enables optimal querying and access to large amounts of data, preventing the Data Lake from becoming a data swamp.
- It provides a historical archive of the source of the data (cold storage), data lineage, auditability, and reprocessing if necessary.

The Silver Layer (Data Filtered, Cleaned and Validated). The data from the bronze layer is filtered, cleaned, trimmed and subjected to other procedures to validate it and is added to the silver layer.

- At this layer, rules are applied to transform the data into the desired format.
- The data is transformed allowing the elaboration of elements such as: ad hoc reports, sophisticated analysis, machine learning.
- The Silver layer unifies data from many sources globally.

The gold layer (Rich Data). The silver layer data is prepared and combined for global use.

- Business and quality principles and rules are applied to such data to convert it into an appropriate format for analysis.
- This layer is made up of data with fewer joins and data models that are more optimized for reading.
- Provides distributed storage and fast access to data.

Figure 12 presents an outline of the data structure of a system based on Lakehouse technology.

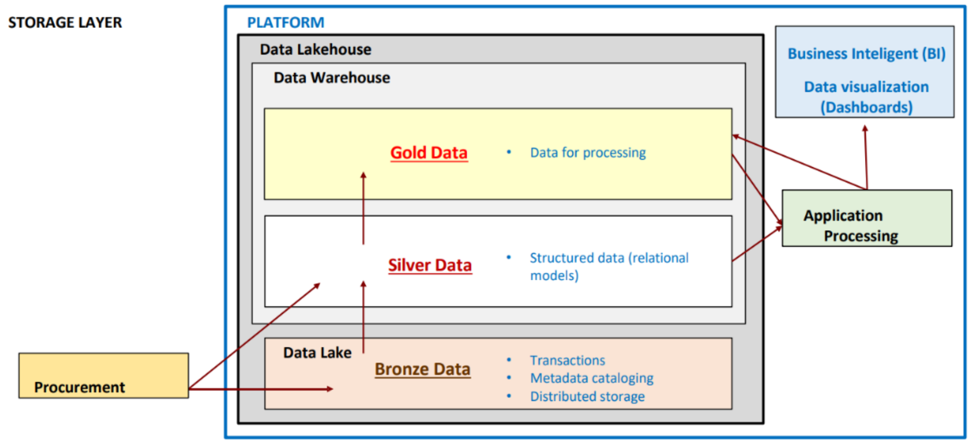

Figure 15. Storage layer schematic.

MINIO

MINIO

The chosen solution for implementing the storage layer for the Data Lake is MinIO. MinIO offers a new storage paradigm based on object storage instead of file-based or block-based storage. This involves storing data in a flat file architecture along with metadata associated with each file. This approach enables faster data retrieval, optimizing resources, reducing costs through easier horizontal scaling, and providing infinite scalability.

The use of a distributed storage component is essential in a project of this nature, where, due to the high level of monitoring that will be applied, a huge amount of data is expected, which will require both distributed storage and processing. Without an architecture that guarantees horizontal scalability, such as the one offered in Data Lake scenarios, there is a risk that when the volume of data in the future platform increases significantly, not only will the infrastructure's capacity be exceeded, but there will also be no simple alternative to increase that capacity.

APACHE HIVE METASTORE

HIVE

It is a solution for managing huge datasets and performing queries on them. It is also used for metadata cataloguing.

- *Processing layer*

It includes the selected technologies for data transformation, cleaning, aggregation, and model processing. Tools need to be incorporated that perform their processing tasks in a distributed manner, so that the workload is shared across the different servers that make up the processing cluster.

TRINO

It is a highly parallel, distributed, and scalable open-source query engine, focused on efficient, low-latency analysis. Trino extends the capabilities of a Data Lake by adding Data Warehouse-like behaviors, fully forming the Lakehouse architecture.

APACHE SPARK

It is a framework for building open-source distributed computing systems. It is very efficient with massive data processing, offering great speed and processing capacity.

- *Services layer*

CLICKHOUSE

ClickHouse is designed to provide high performance for analytical queries using a SQL-like query language to query data and supports various data types, such as integers, strings, dates, and floats. It offers several features, such as clustering, distributed query processing, and fault tolerance. It also supports data replication and sharding. In our case, it will be used for provisioning the necessary data to feed the system created by BIM6D.

MLFLOW

It is an open-source platform designed to simplify the machine learning lifecycle. It provides a comprehensive set of tools and frameworks to manage and track the entire ML development process from start to finish, including experimentation, reproducibility, deployment, and collaboration.

- *Data exploitation, visualization and publication layer:*

It provides the publication of data in an intuitive and engaging way for the platform's end users: decision-makers, analysts, operational users, data analysts, etc. This layer is responsible for providing end users with a "toolbox" that allows them to carry out their work comfortably and efficiently. In the toolbox, we can find functionalities such as the following: Dashboards, Analytical screens, and Data visualization and self-service tools.

SUPERSET

It is an open-source data exploration and visualization platform used in the fields of AI/ML and data science. It features an intuitive interface with powerful analytical capabilities and extensions. It allows users to: connect to multiple data sources, explore data, create interactive dashboards, and share visualizations with others.

CLOUDBEAVER

It is a web-based database management tool designed to simplify the management and access to multiple database systems from a unified interface, accessible through a browser.

- *Data governance layer*

It is about managing the data. To do this, metadata is created to inform about:

- How clean the data is,
- When it was last updated,
- Who is responsible for it and who maintains it,
- Where it is stored,
- What each column or attribute means, etc.

This layer is composed of tools that enable: Data governance, data glossaries, and data lineage. Some alternatives: DataHub, Apache Atlas, Amundsen, OpenDataDiscovery, or OpenMetaData.

OPENMETADATA

OpenMetadata is an open-source metadata repository that facilitates data cataloging, discovery, and collaboration across your data ecosystem, aggregating metadata from all sources and presenting it to the user based on their needs.

- *Tool layer:*

It is the layer that is responsible for providing end users with a "toolbox" that allows them to carry out their work in a comfortable, efficient and reusable way. Among this toolbox we find functionalities as diverse as the following:

- Tools for data management through the Medallion *Architecture*, to assist in transformations of source data, gaining new insights.
- Notebook tools for the creation and testing of *Machine Learning* models, which will later be deployed through the Service Layer.

Technical Solution:

This layer provides end users with the tools that allow them to carry out their work. Among these tools, there are components for building ETLs and ELTs that will form and implement the medallion architecture, such as:

- DBT (DATA BUILD TOOL): Its main goal is to build and manage data pipelines, transforming raw data into ready-to-use data, properly processed and transformed, for analysis in a data warehouse.

- DAGSTER: An open-source platform for orchestrating data workflows and ETL processes.
- JUPYTERLAB: An open-source client-server application that allows users to create and share web documents in JSON format (notebooks). These are used for experimentation while developing a model.
- APACHE AIRFLOW: An open-source tool for creating, scheduling, and monitoring data pipelines. The basic principle of Airflow is to define data pipelines as code, enabling dynamic and scalable workflows. Airflow uses directed acyclic graphs (DAGs) to represent workflows.

Figure 13 presents a comprehensive view of the solution for the architecture of the digital twin, now including all the apps in the different technological platforms that are part of the solution.

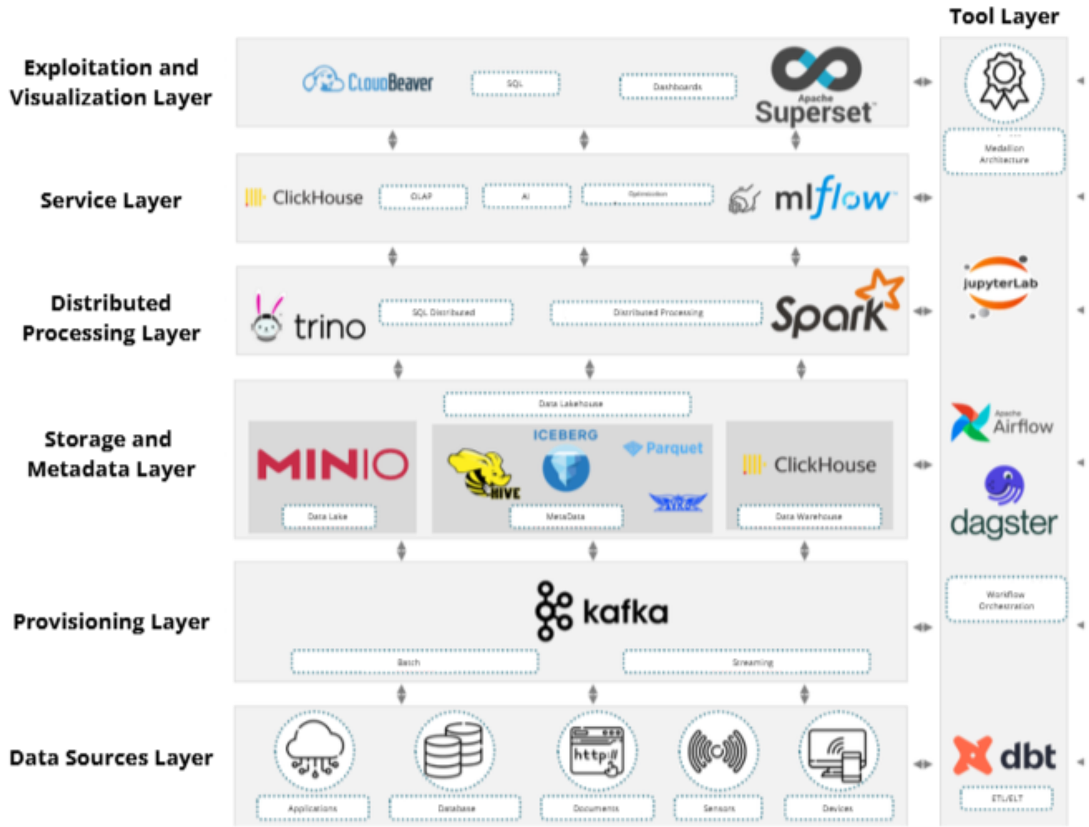

Figure 16. Comprehensive view of the solution for the architecture of the digital twin.

7. ADDING BIM AS AN ASSET REFERENCE SYSTEM DATA MODEL

At the conclusion of the work, which is the background for the comprehensive process digital twin, the following tasks were accomplished:

1. BIM Modeling of the Workshop Building:

   - A BIM model of the workshop building was created and exported in IFC format, a widely accepted standard for interoperability in Building Information Modeling. The IFC model provides a structured and standardized representation of the workshop's physical and functional elements, enabling seamless integration with other digital systems through the Common Data Environment (CDE).

2. Creation of a Digital 3D Representation for the Tamping Machine:

   - A 3D digital twin of the tamping machine was developed in GLTF format, a lightweight and modern 3D file format optimized for real-time applications and web-based visualization. The GLTF model accurately represents the geometry and key features of the tamping machine while ensuring high performance and compatibility with visualization platforms.
   - Key parameters were defined to facilitate the integration of client-provided data into the broader digital twin system through the CDE, ensuring consistency and usability.

3. Integration of the Digital Twin with POWERBIM:

   - The IFC model of the workshop and the GLTF digital twin of the tamping machine were integrated into POWERBIM via the CDE, enabling seamless merging of building and machinery data.
   - The CDE acts as a centralized hub, managing and harmonizing data from multiple sources to ensure interoperability and consistency across the digital twin system.
   - This integration created a highly interactive environment that serves as a reference system for visualizing assets, facilities, and processes. Users can navigate the 3D space and click on any component to instantly access relevant data, such as operational parameters, maintenance history, and assigned workflows.
   - Dynamic data from the CMMS system and DataLake Systems was incorporated through the CDE, enriching the digital twin with real-time, actionable information.
   - Interactive reports were generated within POWERBIM, leveraging the CDE to enhance visualization and provide tailored insights on key operational and maintenance data.

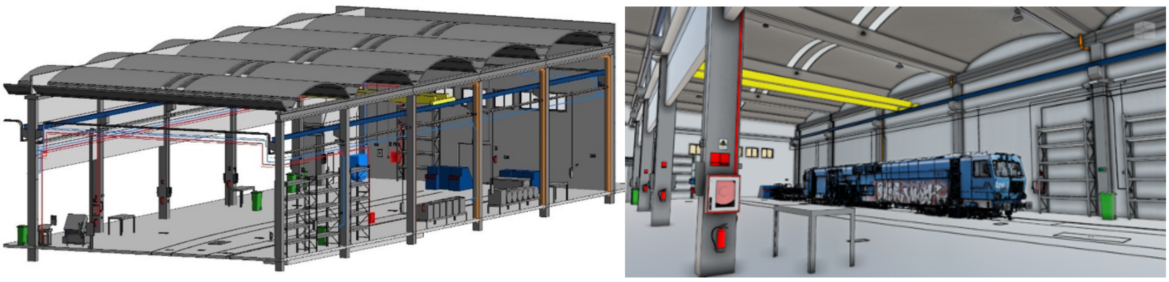

Figure 14.BIM WORKSHOP Model and tamping machine 3D model Integration

7.1. Real-time Virtual Visualization with POWERBIM Cloud Platform

Once the digital representation of assets, facilities, and processes has been created, the system can enable a real-time virtual visualization of the workshop in the cloud. This is achieved by integrating the static representations (e.g., 3D models in IFC and GLTF formats) with real-time transactional data generated by the workshop's operations. How It Works:

1. Data Integration via the Common Data Environment (CDE):
   - Static Data: The CDE centralizes representations of assets and facilities, such as the workshop building and machinery, which serve as a static, visual framework. These models are stored in standardized formats like IFC (for

facilities) and GLTF (for machinery), ensuring interoperability and high-performance visualization.
  - Dynamic Data: Real-time transactional data, including inspection results, operational statuses, and maintenance workflows, is continuously collected from CMMS systems, IoT sensors, and other data sources. This data is processed and stored in a Data Lakehouse, ensuring efficient retrieval and analysis.
  - By integrating static and dynamic data in the CDE, the platform enables seamless coordination between the digital and physical elements of the workshop.
2. Cloud Platform Integration:
  - The 3D digital twin of the workshop is hosted on a cloud platform such as POWERBIM, which merges the static 3D models from the CDE with dynamic data streams.
  - APIs and real-time data pipelines (e.g., Kafka or Spark Streaming) facilitate synchronization between physical workshop activities and their digital representation, ensuring data consistency and real-time updates.
3. Real-Time Visualization and Interaction:
  - Users access the virtual workshop through an interactive cloud interface, where the system overlays real-time data onto static 3D models. This allows visualization of operational states, including:
    - Status of machinery (e.g., active, under maintenance, idle).
    - Ongoing inspection results displayed as interactive overlays.
    - Location and activity of personnel within the workshop.
  - Visual cues, such as color-coded elements or notifications, highlight key issues or operational statuses.
4. Remote Monitoring and Control:
  - The virtual environment allows users to navigate the workshop remotely and interact with its components for detailed inspection. For instance:
    - Clicking on a machine provides its maintenance history, operational parameters, and inspection results.
    - Selecting a specific facility zone reveals environmental conditions, personnel activities, and task progress.

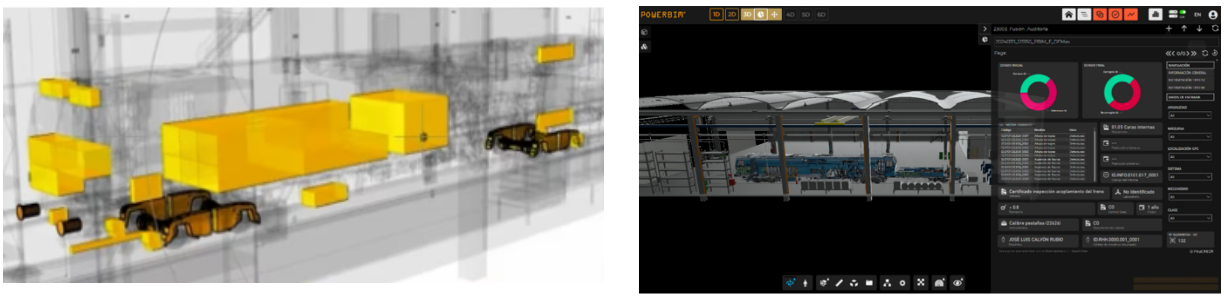

Figure 15. POWERBIM Added Capabilities

7.2. Benefits of the System

This integration of static digital models with dynamic real-time data provides a powerful tool for workshop management, combining **visual clarity** with **actionable insights**. By visualizing the workshop's current state in real time, stakeholders gain unparalleled control and oversight, enhancing both operational efficiency and strategic decision-making. More precisely they gain:

- **Operational Transparency**: Users can monitor the entire workshop's status in real-time without being physically present, enabling **remote management** and decision-making.
- **Efficiency in Inspections**: The results of inspections for different systems are immediately visible, helping users identify and address issues without delay.
- **Predictive Insights**: By connecting real-time data to predefined **data models** and **AI algorithms**, the system can generate predictive maintenance alerts and operational recommendations.

POWERBIM

POWERBIM is a cloud-based platform that integrates Building Information Modeling (BIM) with real-time data from IoT sensors, CMMS systems, and data lakes to create dynamic, interactive digital twins. It enables seamless visualization, analysis, and management of facilities, assets, and processes in a unified 3D environment. With POWERBIM, users can remotely monitor operations, access actionable insights, and optimize decision-making through an intuitive interface.