

A Big Data implementation of the MANTIS Reference Architecture for Predictive Maintenance

Felix Larrinaga¹, Javier Fernandez¹, Ekhi Zugasti¹, Iñaki Garitano¹, Urko Zurutuza¹, Jon Olaizola¹
Mikel Anasagasti² and Mikel Mondragon²

Abstract—This paper presents the implementation of a reference architecture for Cyber Physical Systems (CPS) to support Condition Based Maintenance (CBM) of industrial assets. The article focuses on describing the Data Analysis approach to manage predictive maintenance of clutch-brake assets fleet over the previously defined MANTIS Reference Architecture. Proposals for both the architecture and Data Analysis implementation supports working on Big Data scenarios, due to the usage of related technologies, such as: HDFS, Kafka or Apache SPARK. The techniques are (1) Root Cause Analysis (RCA) powered by Attribute Oriented Induction (AOI) Clustering and (2) Remaining Useful Life (RUL) powered by Time Series Forecasting. The work has been conducted in a real use case within the H2020 European project MANTIS.

Index Terms—Keywords-Industry 4.0; Reference Architecture; Condition Based Maintenance; Cyber Physical Systems

I. INTRODUCTION

The proliferation of CPS, Internet of Things (IoT) and cloud technologies are opening new opportunities for collaboration between systems, platforms and applications. This trend affects all aspects of life and numerous domains (Smart Cities, Energy, Health, etc.). Industry is not alien to these changes and opportunities. Manufacturers, production lines, solution integrators and engineering companies are investing to update their machinery and systems to the new situation.

Maintenance is essential for improved performance of industrial assets and processes. While reactive maintenance focuses on repairing an asset once failure occurs, proactive maintenance focuses on avoiding repairs and asset failure through preventive and predictive methods. CBM is a predictive maintenance strategy that is based on the continuous monitoring of various parameters of an asset to evaluate its health level and future performance.

The availability of large quantities of data through IoT and CPS triggers the implementation of advanced monitoring strategies for asset management while facilitating the adoption of policies and strategies for maintenance measures, such as CBM [1]. However, this raises higher level issues that might have not been considered previously for condition monitoring scenarios: 1) How to transmit these data from the physical system and to where? 2) How to create interoperable

data representation and semantics? 3) What can be the backend that processes this inbound data streams in a scalable manner? 4) How can we still maintain real-time restrictions and abide by communicational constraints? Even though data might be available with great time and value resolution, it is often not practical to be transmitted "as is" from the device or machine for communicational constraints.

To address these issues the ECSEL Project MANTIS [2] was born in 2014. In a paper published by C. Hegedus et.al.[5] the MANTIS framework was proposed. F. Larrinaga et. al. [28] implemented the architecture in a real case. Also, in MANTIS the results of other use cases were analyzed. Some of the implementations were exposed in the paper published by E. Jantunen et. al.[27], and shows a comparative study between the approached implemented.

This paper presents an extension of the MANTIS reference architecture for the use case presented in [28] (GOIZPER use case) but focusing on the data analysis techniques used and the results obtained. First, the background of the project and the use case are presented (section II). Section III introduces the MANTIS reference architecture. Section IV outlines the technologies and tools used to implement the reference architecture. Section V and VI present the Data Acquisition and Edge Computing section, and the Data Interoperability section. In section VII the explanation of the algorithms for CBM implemented in the use case at platform level are exposed. Finally, results and conclusions are presented in Sections VIII and IX.

II. BACKGROUND

A. Maintenance

According to R. Keith [18] there are three categories of Maintenance: (i) Preventive Maintenance (aka Time-Based Maintenance (TBM)) tries to establish periods of action in order to perform the maintenance tasks. (ii) Corrective Maintenance pretends to manage and schedule reparations after a failure has occurred. (iii) Predictive Maintenance is focused on preventing unscheduled downtimes and premature damage on the equipment. Usually it is considered another maintenance strategy called Proactive Maintenance (iv), which works correcting the root causes of failures in order to extend the lifetime of the machine.

The implementation of Corrective maintenance does not imply high IT inversions to measure availability, performance, etc. The investment cost is low. On the other hand the availability of production equipment, the efficiency of production and the quality of production are lower than

*This work was not supported by any organization

¹Electronics and Computing Department, Mondragon University, Loramendi 4, 20500 Mondragon, Spain flarrinaga, jfernandez, ezugasti, igaritano, uzurutuza, jolaizolaa@mondragon.edu

²Goizper S.Coop., Barrio Antigua, 4, 20577 Antzuola, Guipúzcoa masasagasti,mondragon@goizper.com

is the case with more sophisticated maintenance strategies. When introducing Preventive Maintenance, or Time-Based Maintenance, the idea is that maintenance is carried out prior to such failures that can abruptly stop the production. This in practise means time/calendar based maintenance. Even though Preventive Maintenance is in many cases an improvement to Corrective Maintenance it is not an ideal approach as it can easily lead over maintaining i.e. such equipment are also repaired or maintained that actually do not need maintenance. The worst scenario here is that in some cases maintenance can be the cause of failures so a machine that would otherwise work might stop due to maintenance. In Predictive Maintenance condition monitoring is used to support decision making i.e. some methods are used to determine whether the machine needs maintenance and what would be the optimal time to do this. In principle when following the Predictive Maintenance strategy the idea is to carry out maintenance for high number of components at the same so that the number of stoppages of the production machinery can be minimized. In Proactive Maintenance strategy the idea is to be able to reduce the need for maintenance in the first place. In order to do this it is necessary to be able to understand what is causing the wear of machinery and what could be the ways to avoid this.

Although Predictive and Proactive Maintenance have similar characteristics, they differentiate in some aspects. Predictive maintenance systems are characterized by monitoring for the purpose of detecting wear and malfunction before failure with following downtime and costly repairs. This means to identify without manual inspection when parts need to be replaced and may involve adapting usage to ensure operation until next service opportunity. This includes making sure that the needed spare parts are manufactured and delivered to be available at the intended maintenance occasions, preferably just in-time. One major challenge is related to the huge amount of data that to large extent contains imprecise information.

Proactive Maintenance commissions corrective actions aimed at sources of failure. It is designed to extend the life of mechanical machinery as opposed to (i) making repairs when often nothing is broken, (ii) accommodating failure as routine and normal, and (iii) preempting crisis failure maintenance - all of which are characteristics of the predictive/preventive disciplines.

A full categorization of Maintenance methodologies is shown in Figure 1.

Three important concepts determine Predictive Maintenance.

(i) Anomaly Detection (AD): the aim is to find unusual behavioral states in the monitored asset that can be considered as anomalies. In the last decade, fault detection systems have become a very important and active field in industry. With the help of the multiple technology improvements many enterprises have the possibility to install sensors in their machines or machine components, and thus, the capability to monitor their health-status. This implies an important improvement at the time to tackle the maintenance for assets,

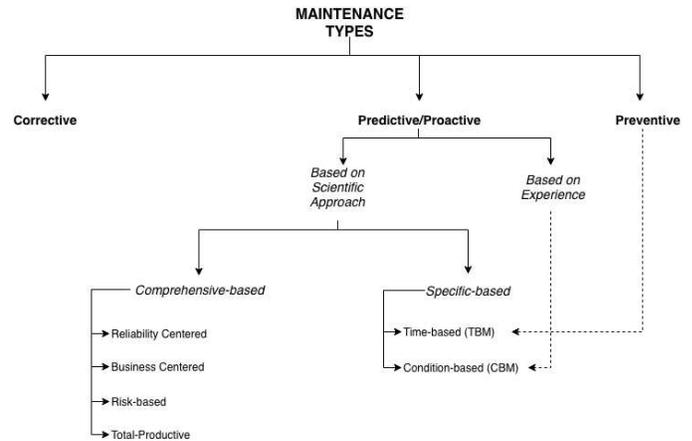


Fig. 1. Global scheme of the maintenance methodologies

since sensors they made possible to generate alert systems that helped detecting anomalies. Therefore, they can optimize the actuations for correcting and repairing.

In [19], a study of model-based anomaly detection is presented, where Venkatasubramanian, V. remarks the importance of the *a priori knowledge*. This is a knowledge defined previously by domain experts. This and other papers such as [20], [19], [21], [22], [23] insist on the importance of existing a background knowledge, explicitly defined, which represents the different anomaly and fault states in order to determine the causality of an error.

(ii) Root Cause Analysis: the goal is to describe the reasons why an anomaly has occurred, as a specification or refinement of the AD step. In order to perform the analysis of calculation of both AD and RCA, many machine learning algorithms are proposed, such as: Bayesian Networks, Support Vector Machines (SVM) or Artificial Neural Networks (ANN).

In 2016, Chemweno, P. [20] proposed a novel method to estimate RCA, based on different data mining techniques to manage process-steps. The motivation of this study is that most commonly used algorithms in RCA such as Bayesian Networks and Fault Tree Analysis, depends mostly on the cause-effect associations defined by domain experts in the domain knowledge, delinked from failure associations embedded in the empirical failure events. Thus, in this study the author utilizes four different Machine Learning clustering methods, in order to group the different failures with similar characteristics in different representative groups. This way, the idea is to facilitate and optimize the analysis resizing the failures defined in the knowledge-base according to their similarity/dissimilarity.

Finally, (iii) Remaining Useful Life: the idea is to define the resting time until the next breakdown will produce. In [24] Ahmadzadeh, F. and Lundberg, J. make a review of the different techniques to estimate RUL, and distinguishes four different methodologies: (i) *physics based models* are oriented to construct technically comprehensive theoretical models to describe the physics of the system; (ii) *experimen-*

tally based models, uses probabilistic or stochastic models of the degradation phenomenon or the life cycle of components by taking into account the data and the knowledge accumulated by experience; (iii) *data-driven models* are founded on the processing of the collected data, not really needing special product knowledge to be specified; and (iv) *combination/hybrid prognostic methodologies*, which is based on combining more than one previously mentioned models.

B. Use Case

The main objective of MANTIS is to develop a Cyber Physical System (CPS) based Proactive Maintenance Service Platform Architecture enabling Collaborative Maintenance Ecosystems. This reference architecture has to fulfil the requirements established by several industrial use cases demanding solutions for CBM that include sensors and SW at CPS level as well as a platform and tools for data analysis.

Intensive work has been conducted in the project to conceptualize, define and design the reference architecture and to identify the components, technologies and tools necessary to implement such solution. Those components, technologies and tools depend in great extent on the requirements established by the use cases in the project.

One of those use cases in MANTIS is concerned with analysing the clutch brake system and its components in press machines to detect the most important failure sources and be able to perform predictive maintenance in those press machines. The use case is led by GOIZPER. GOIZPER is one of the market leaders of power transmission components for metal forming machine tools like clutches, brakes or cams. Their final customers come from highly demanding sectors such as all world's top automotive manufacturers, stampers, home appliances or metallic furniture and are demanding products with high levels of quality and availability seeking a drastic reduction of high cost caused by production downtimes with required maintenance-repair operations and a better delivery times' compliance. That is why GOIZPER considers critical to increase machines and components reliability. To meet this challenge GOIZPER wishes to incorporate cutting-edge technologies in their products as a means of enhancing products robustness and functionality in order to facilitate proactive-predictive maintenance activities.

In MANTIS, the use case considers a test bench containing a Clutch Brake component. The use case has mainly incorporated dedicated smart sensors, pre-processing and data acquisition, communications in harsh environments, a platform for data gathering, treatment and analysis as well as tools and algorithms to support decision-making and to identify problematic situations.

The clutch brake contains two friction discs, which are the elements to transmit power to the system. One of the friction discs is totally stopped (attached to the machine static part). The second friction disc is always rotating at the machine nominal speed (non-stop friction disc). So when the clutch brake piston makes contact with the first friction disc surface, the clutch brake and the whole system after it will be stopped immediately (0rpm). However, if the clutch brake piston

makes contact with the second friction disc surface the whole power transmission system after the clutch brake will start rotating at nominal speed. This friction discs material get wear while they are used, as in conventional bicycle brakes. Several parameters are very important to check the friction disc wear status.

- Friction disc position: position of the rotatory friction disc and output shaft.
- Springs force: springs are responsible to push the clutch brake piston towards the stopped friction disc or towards the rotating friction disc.
- Friction disc wear: this is the attribute that we must calculate, in order to predict when it will run out of material.



Fig. 2. GOIZPER use case test bench.

The overall objective seek by GOIZPER is to early detect internal wear of a GOIZPER clutch-brake. To do that, the moving parts of the clutch-brake need to be sensorized. By continuously monitoring the system conditions proper operation of the clutch-brake can be ensured. Moreover, the most critical operating variables can be registered in the platform in order to analyse the working process and prevent misuses.

In this paper we propose the analysis of parameters and calculated values at cloud/platform level. To achieve this a reference architecture, data acquisition, data interoperability and data analysis are necessary. For AD and RCA, the clustering method denoted Attribute Oriented Induction was used to perform the analysis, with the aim of discovering new unusual or abnormal states in the behaviour of the machine. And for the calculation of RUL, Time Series based analysis was the choice. The following chapters present the approach we followed for the use case.

III. MANTIS REFERENCE ARCHITECTURE MODEL

MANTIS Architecture Reference Model (MANTIS-ARM) has been created to provide the cornerstone for designing, developing and deploying Predictive and Preventive Maintenance MANTIS-enabled architectures and solutions. The MANTIS-ARM consists of five main elements:

- Reference Model: a reference model is an abstract framework for understanding significant relationships among the entities of some environment [3].

- Reference Architecture: provide a template solution for the architecture (aka. architectural blueprint) for a particular domain [4].
- Feature model: Introduces key concepts to characterize common and varying aspects in the architectures to be derived from the reference architecture.
- Guidelines: discusses how the provided models, views and perspectives are used.
- Reference applications: show the diversity of the included solution variants, and thus illustrate architecture signification features and related design decisions.

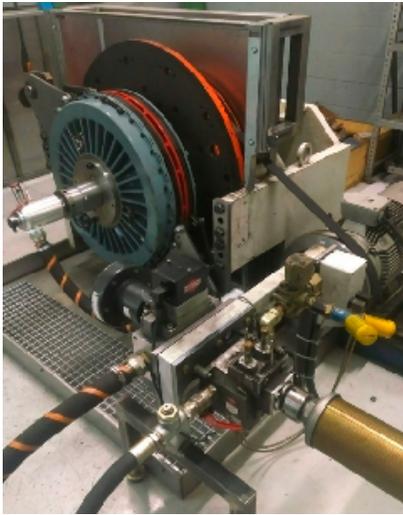


Fig. 3. GOIZPER use case clutch-brake in operating mode

The mission of the platform architecting activities in MANTIS includes to devise the overall architecture of the MANTIS distributed system for proactive maintenance, to address issues that have an impact on several steps in the chain of turning raw data into information usable for distributed decision-making and to consider key aspects: interoperability, consistency, availability, reliability, robustness, safety and security of the system as a whole.

A reference service platform architecture shall allow for industries participating in MANTIS to take advantage of progress on proactive maintenance in related but different industries. In addition, it allows for less mature industries in the project to reuse experiences from industries in the forefront of proactive maintenance. They can thereby ensure that improvements in maintenance can be achieved gradually and consistently with future plans and best practice. Important aspects that the architecture should address towards the use cases are:

- Interface, protocol and functional interoperability
- Data validation ensuring that data analyses are made on data that give clean, correct and useful data information about the system.
- Distributed data, and information processing and decision-making ensuring consistent behaviour and avoid contradicting actions, e.g. between local and distributed data analysis and decision-making.

- Information validation ensuring that created information still is relevant for the system analysed in particular for CBM.
- Safety and fault tolerance ensuring that critical information remains available and following decisions can be taken or proposed although partial system failure.
- System and service level security ensuring that the system incorporates means to hinder misconfiguration and can be protected from wire-tapping and various attacks.
- System engineering and reusability of defined and existing services.
- System verification and validation of the service platform architecture and overall design, covering both functional and non-functional properties.

A. Architecture and Interoperability levels

The Mantis architecture described in [5], considers a number of components separated in three tiers. The identification of those levels for interoperability is inspired by the IIC-RA three-tier architecture pattern [6] that comprises edge, platform and enterprise tiers (see Figure 4). These tiers play specific roles in processing the data flows and control flows involved in usage activities.

The edge level comprises physical entities that belong to the same local network and or functional area. It implies the virtualization of the physical entities into CPS (called component level interoperability) and the provision of the data extracted from the CPS to the platform level. The platform level receives processes and forwards commands from/to the edge level. It provides more complex and resource consuming data analytics and knowledge generation functionalities. It is also concern on how to represent the knowledge models generated by data analytics digital artefacts. The enterprise level is concerned with the applications that integrate information from one/several sites to enhance the global decision-making process using monitoring through Human Machine Interfaces (HMI) and data aggregation and analysis.

B. Methodology

The methodology followed to build the GOIZPER architecture, components and its data models is also specified in Mantis (see Figure 5). The approach for designing MANTIS use cases follows the principle of architecting for concrete stakeholder concerns (Architecture Drivers). Architecture drivers consider business needs as well as functional and quality requirements. Constraints limit the scope of the solution to build. These stakeholder concerns will drive the actual architecture design, which is based in the approach follow by the SPES consortium [7]. This approach suggest to start by delineating system and its context. Continue with the functional decomposition of the system. The next step is the software realization of software. The final steps considers the hardware realization of functions and the deployment of software entities.

IV. PLATFORM TECHNOLOGICAL IMPLEMENTATION

The implementation of the Mantis reference architecture in the GOIZPER use case constituted the technological platform shown in Figure 5, which implements the following blocks:

- 1) Data Access and Ingestion through the Edge Broker: Provides the access to the platform and the adaptation of the messages coming from the CPS to the Information Models and data structures provided at platform level. The Edge Broker is composed of:
 - a) Publish-Subscribe servers: manage messages from/to CPS and internal components of the platform using queues and exchanges. The solution has been implemented using RabbitMQ [8] and Advanced Message Queuing Protocol (AMQP) [9].
 - b) Translator/Converters: convert or translate input data formats into output data formats and perform protocol mapping. CPS messages are generated according to the MANTIS Event Information Model (based on the IoT-A event information model) and converted into the storage formats presented in the next paragraph. The converters have been implemented using an Enterprise Service Bus (ESB) named WSO2[10].
- 2) Data Storage systems: store information coming from CPS and results of data analysis maintenance actions. Two storage systems:
 - a) MIMOSA DB[11]: is a database compliant with the ISO-13374 Standard (Condition Monitoring and Diagnostic of Machines). According to this standard, a CBM system should be composed of various functional blocks: Data Acquisition (DA), Data Manipulation (DM), State Detection

(SD), Health Assessment (HA), Prognostics Assessment (PA) and Advisory Generator (AG)[12]. One of the main objectives of the MIMOSA CBM architecture is to standardize the information flow between the various blocks, so that equipment from different vendors could be interoperable. The MIMOSA database is deployed in SQL Server and API REST is used to access data from applications.

- b) Hadoop Distributed File System (HDFS): is a distributed file system designed to run on commodity hardware. Designed to be deployed on low-cost hardware, HDFS is highly fault-tolerant and provides high throughput access to application data, which makes suitable for applications that have large data sets.
- 3) Batch Processor: data analysis and processor mechanisms to enable the management of large volume of data, fetched from storage systems and process on demand. Implemented using Apache Spark [13]. See next section for details on data analysis techniques employed in the use case.
- 4) HMI: Describe purpose and technological solution.

V. DATA ANALYSIS AND COMPUTING AT EDGE LEVEL

Edge computing approaches open the way for directly or indirectly measuring system magnitudes that will feed the cloud. These approaches also provide a means for acquiring knowledge about the physical system's performance in place.

Model Based Soft Sensors (MBSSs) [25] have been applied in order to monitor the condition and the dynamic behaviour of the clutch brake. Two components of the clutch brake have been monitored through a MBSS algorithm: the condition of brake springs and of the friction discs.

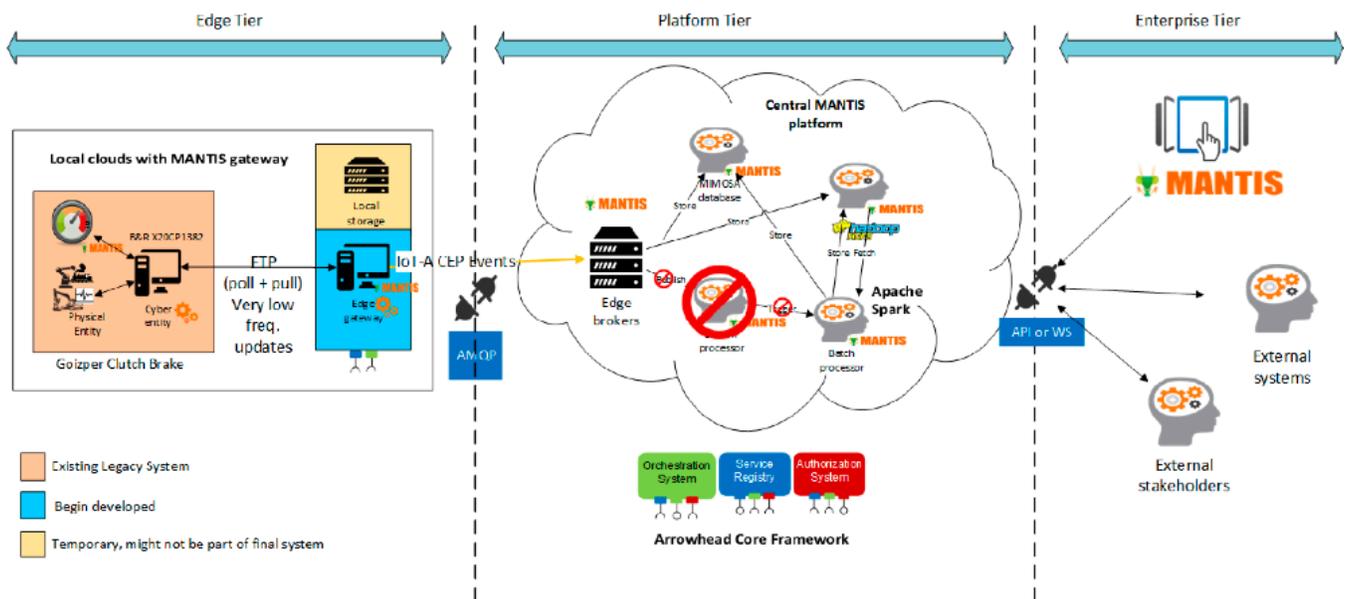


Fig. 4. Mantis Architecture levels.

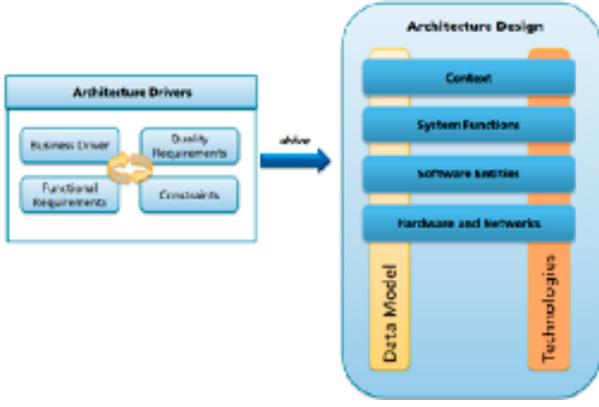


Fig. 5. Mantis architecture construction approach.

The brake springs within the clutch brake create a constant force against the brake side, keeping the component static and the press stopped when it is not powered. Brake springs and their location are depicted in Figure 7.

Friction material discs are the power source parts of the clutch brake, both connected to an external element. There are two friction material discs. The one located at the clutch side is attached to the flywheel (electric motor powered). The other one is situated at the brake side and it is joined to a static part of the press machine. Clutch side friction disc is constantly rotating, meanwhile brake side's is a static component. A friction disc is shown in Figure 8.

The proposed MBSS algorithm is able to estimate the condition of both components taking advantage of the already installed pressure sensors placed by factory default. The algorithm requires the mathematical representation or model of the clutch brake as well. The developed model encompasses both, a pneumatic subsystem and a mechanical one which describe the behavior of the clutch brake, as shown in Figure 9.

A. MBSS design Model

The clutch brake and the pneumatic system that powers it have been modeled. Equation 1 represents the mechanical subsystem.

$$\ddot{x} = F(p) - M(x, \dot{x}) \quad (1)$$

Where x , \dot{x} and \ddot{x} represent respectively the position, velocity and acceleration of the piston of the clutch brake, $F(\cdot)$ expresses the input force that depends on the pressure p of the clutch brake chamber and $M(x, \dot{x})$ is the mechanical subsystem that comprises the mass of the piston and the elastic constant of the brake springs.

At the same time, the pressure of the clutch brake chamber is obtained by means of the line pressure and the electrovalve output pressure, required to feed the pneumatic subsystem of the model defined in equation 2.

$$\dot{p} = Pn(x, \dot{x}, p_1, p_2) \quad (2)$$

Where x and \dot{x} are piston position and velocity respectively and p_1 and p_2 are line pressure and electrovalve output pressure. $Pn(x, \dot{x}, p_1, p_2)$ denotes the pneumatic model and \dot{p} is the chamber pressure differential over time. The mechanical and pneumatic subsystems are coupled through p , x and \dot{x} .

Estimation of the parameters A Bayesian MBSS has been used in order to estimate the condition of the both elements described before. The friction material thickness and the stiffness of the brake springs do not change rapidly over time, so they are considered as parameters of the model [26]. For that reason a batch MBSS has been selected, which performs the estimation after the whole test have been recorded. Figure 10 shows the flow diagram of the applied MBSS.

B. Validation. Model/algorithm and data

Spring stiffness:

The developed soft sensor is able to estimate the brake springs stiffness (elastic constant) from two pressure measurements applying the aforementioned MBSS. The set-up is shown in Figure 11. The pressure sensor P1 measures the line pressure, the pressure sensor P2 measures clutch brake input port pressure and the pressure sensor P3 measures the chamber pressure (only for validation purposes, it does not participate in the model).

Figure 12 shows the results of two experiments where two distinct brake spring sets were placed, getting different actual values as marked with the red dashed lines.

Numerical results are shown in Table I.

TABLE I
EXPERIMENT RESULTS.

Experiment	Actual Value (N/m)	Estimated Value (N/m)	Relative Error (%)
15 springs	2.82×10^5	2.846×10^5	0.922
10 springs	1.952×10^5	1.931×10^5	1.076

Discs friction material wear:

In this approach two estimated magnitudes have been used in order to relate them with the wear of the friction material of both sides: the air mass flow and the pressure level inside the chamber. This development relates the accumulated air mass flow (i.e. air mass) with the chamber pressure value during the whole experiment.

Many tests have been carried out combining different wear level friction material for both sides. Analysed metrics have revealed a similar behaviour for an identical friction material wear in the same side, either in clutch side or brake side. The setup for the experiment is described in Figure 11.

Figure 13 shows the air pressure vs air mass curves for different friction material wear combinations. The percentages that appear in the figure legend represent the wear level of both sides, being left side percentage brake side wear and right side percentage clutch side wear.

In Figure 14 brake and clutch related curve sections have been zoomed in. It can be noted how the curves track the same path for an identical wear level in both sides.

From these results the wear level of the friction material can be assessed. During the lifecycle of friction discs, their friction material wears making the piston position change. Therefore, it allows more air to enter in the chamber for a certain pressure level comparing to a non-degraded friction discs scenario.

Edge computing solutions provide not only a means to estimate and detect deviations in the condition of the monitored machine on-site, but also to carry out the first filtering of raw data acquired by sensors. Thus, the obtained results are used to feed the cloud platform optimizing the bandwidth (the amount of data transmitted) and giving rise to the detection of a first cause-effect association between the measured signals and the estimated condition of the machine.

VI. DATA INTEROPERABILITY

In this section the parameters selected for the interexchange between the edge and the platform tiers are analyzed.

A. Data Sets

The data extracted from the sensors of the clutch-brake to complete the analysis are all numerical.

In order to perform the calculations, a few additions were done over the collected data, and two more features were added to the structure:

(i) An identification-point feature was set to the data structure. Data is collected in multiple point batches. Each batch is composed by a thousand points, referring to a single clutching or braking process. So, each data point will be the identification of the point number into the batch, from 1 to 1000.

(ii) The real duration of the braking or clutching process is added to the data structure. Although the collected data-batches are composed of 1000 points each, the real duration of the clutching or braking process is not 1000. Actually, there are a few points at the beginning and at the end of the signal which are related to the machine in a resting mode, so, a preprocessing step is performed in order to delete those elements, and only taking the working signal part into consideration for the analysis.

The final structure of the data is shown in Table II, in which the colored cells reflects the new feature additions.

Then, the different monitored attribute are briefly explained.

- Point: Each data-element number (order) of the clutch or brake cycle.
- Duration: The number of points composing the clutch or brake cycle.
- Trigger: Signal representing if the electrovalve is activated or not.

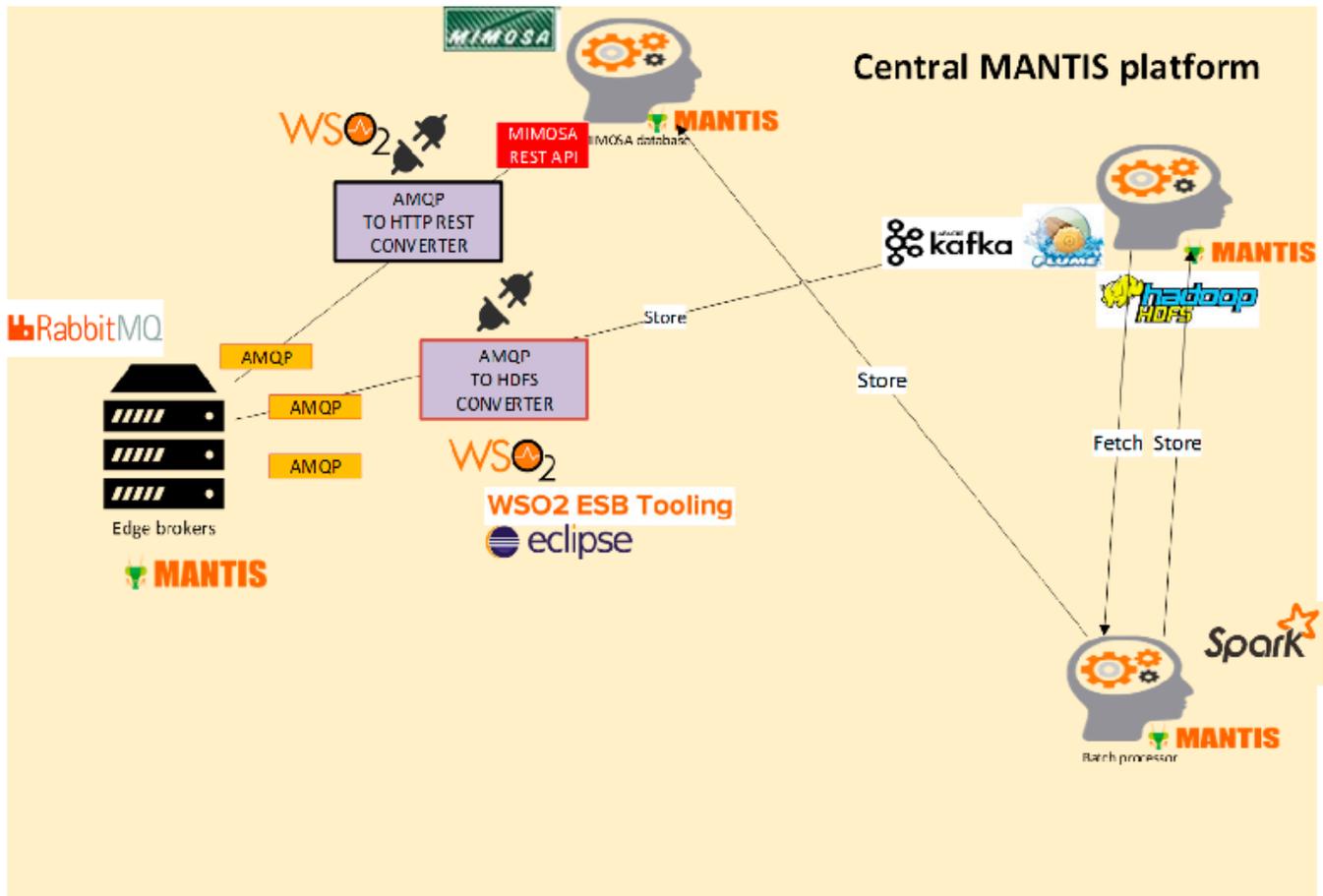


Fig. 6. Goizper platform architecture.

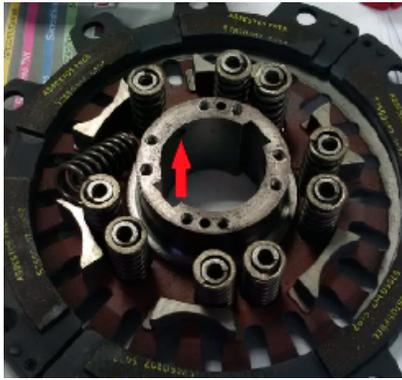


Fig. 7. Brake springs of the clutch brake.

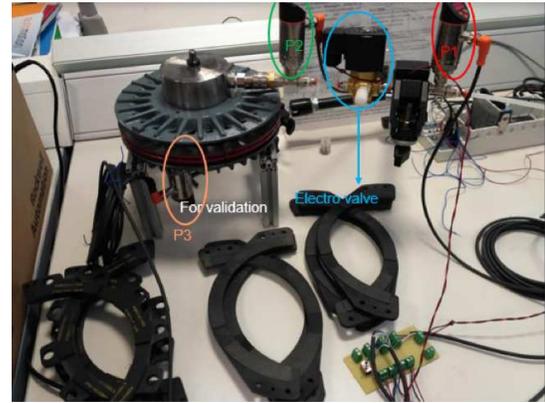


Fig. 11. Prototyping clutch brake set-up.



Fig. 8. Friction disc of the clutch brake. Friction pad is highlighted by a red circumference.

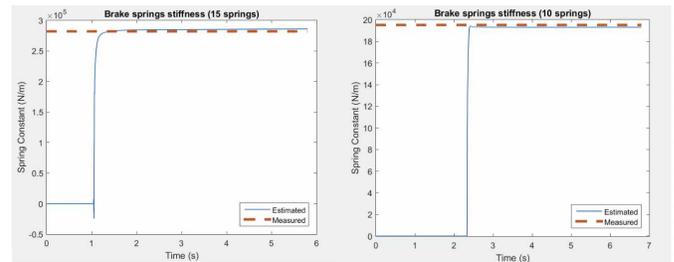


Fig. 12. Estimated vs actual inner chamber pressure and brake springs stiffness.

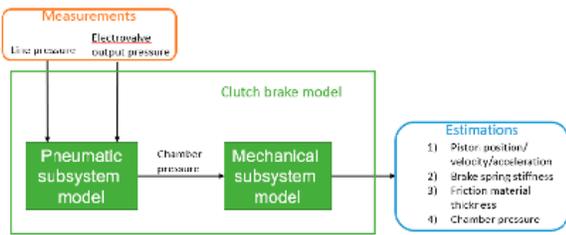


Fig. 9. Clutch brake model diagram.

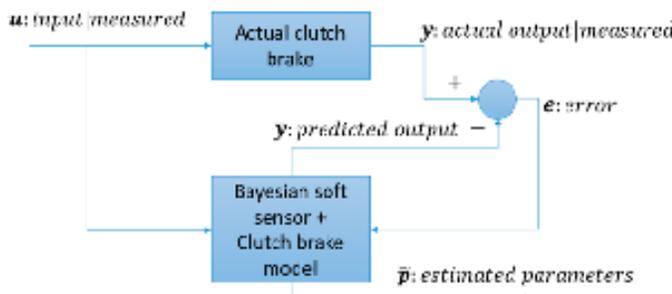


Fig. 10. Batch Bayesian soft sensor estimation performance diagram.

- Application Pressure: Pressure applied to the machine to perform a clutch or brake process.
- Shaft Speed: Speed of the crankshaft.
- Line Pressure: Constant pressure value applied to the machine to perform some clutch and brake characteristics.
- Position: Angular position (degrees) of the rotating part of the crankshaft.

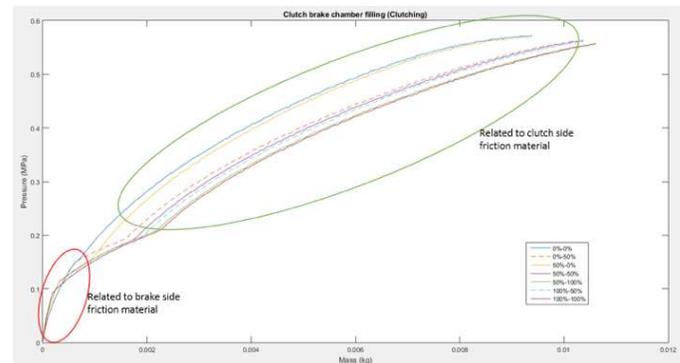


Fig. 13. Air pressure vs air mass representation.

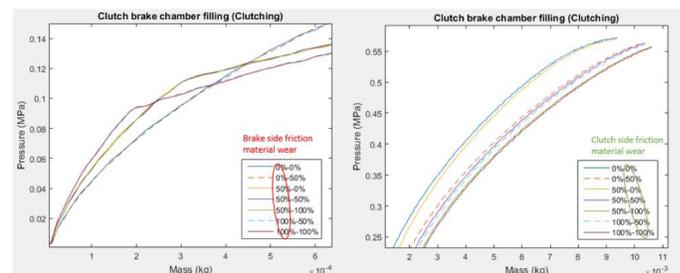


Fig. 14. Left side, brake friction material related curve section. Right side, clutch friction material related curve section.

- Flywheel Speed: Speed of the flywheel.

TABLE II
FINAL DATA STRUCTURE

Point	Dur.	Trigg.	App. Press.	Sh. Spd.	L. Press.	Pos.	Fly. Spd.
-------	------	--------	----------------	-------------	--------------	------	--------------

B. Interlayer Data Exchange and Storage

This section is concerned on how data is exchanged between layers and stored at platform level. Data is exchange among different systems and layers. Physical measurements collected by physical entities are converted into data at component level creating CPS. The edge level is concerned with the CPS belonging to the same local system. The information produced by CPS at edge level is used to model and analyse the behaviour of the system locally but also can be sent to the cyberspace for further analysis. At platform level, data coming from the edge level is translated and organized in order to be processed by digital artefacts.

Data between the edge and platform layers are inter-exchanged using messages triggered by events. Events are significant actions, incidents or episodes that need to be registered and stored in the platform for monitoring purposes or for performance improvement by means of data analysis. Those events store data relevant to the actual process but additionally must collect both spatial and temporal information and associate them to entities/system. Events should be created after one of these situations; sensor reading, operational action, breakdown or maintenance action. For the Clutch-Break system, events are triggered in normal function after clutch and brake operations.

VII. DATA ANALYSIS FOR CBM AT PLATFORM LEVEL

A. Definition of Main Terms

The following aspects of CBM have been addressed for the Goizper use case: (i) Anomaly Detection; (ii) Root Cause Analysis; and (iii) Remaining Useful Life calculation.

The Anomaly Detection is the first and necessary step to identify the main equipment failure causes. The goal is to determine if a current working state is considered normal or not. Then, the Root Cause Analysis step is performed. The aim of RCA is to describe the possible reasons why a current abnormal working state has been produced. Finally, the Remaining Useful Life is used to estimate the resting time until the next anomaly or breakdown can occur. Within GOIZPER use case, Attribute Oriented Induction algorithm has been used as the principal algorithm to calculate these metrics.

Attribute Oriented Induction algorithm is considered a hierarchical clustering algorithm. First proposed by Jiawei Han et al. [14] as a method for knowledge discovery in databases, it is currently considered as a rule-based concept hierarchy algorithm. The representation of the knowledge is structured in different generalization-levels of the concept hierarchy with IF-THEN rules. The execution of the algorithm AOI follows an iterative process in which each variable (also referred as attribute) is generalized based on its own

hierarchy-tree. This step is denoted as concept-tree ascension, by Cheung et. al. [15]. To ensure the correct functioning of the algorithm, it is necessary to establish background knowledge, which specifies attribute generalization levels. That background knowledge was specified by the domain experts of the clutch-break, who know strongly the different characteristics of the features of the machine.

Table 1 shows a visual representation of the generalization process. The first step is to select the variable with the higher number of distinct values, 39 in the example, to then generalize following the criteria established by the background knowledge (e.g. [0, 3]: X, (3, 100]: Y).

By identifying data similarity clusters, AOI provides a knowledge representation of different machine behaviour states. The generation of machine behaviour states knowledge base during the learning process, ensures the representation of all possible machine-working states. Hence, the anomaly working states detection step is simplified, since it only requires the identification of a previously unknown machine behaviour state. This step is called Detection.

TABLE III
RESULTS BEFORE GENERALIZATION STEP

	Var 1	Var 2	...	Var N
Tuple 1	0	0.50	A	21
Tuple 2	1	0.90	A	24
Tuple 3	1	1.80	A	30
Tuple 4	1	2.40	B	38
Tuple 5	1	4.20	C	42
Tuple 6	1	4.80	C	43
...
Tuple 39	1	3.21	F	39
# Dist. Val.	2	39	6	30

TABLE IV
RESULTS AFTER GENERALIZATION STEP

	Var 1	Var 2	...	Var N
Tuple 1	0	X	A	21
Tuple 2	1	X	A	24
Tuple 3	1	X	A	30
Tuple 4	1	X	B	38
Tuple 5	1	Y	C	42
Tuple 6	1	Y	C	43
...
Tuple 39	1	Y	F	39
# Dist. Val.	2	2	6	30

B. Methodology

To perform the Anomaly Detection, the AOI algorithm is launched over the data related to correct working states. This way, many clusters referred to normal behaviour are generated taking into account the values of the attributes and the knowledge specified by domain experts previously. So the generated knowledge base will store clusters referring to normal behaviour of the machine. But at this point, a previous processing step must be considered before starting

the AD calculation: *Quantification* (also referred in the study as Normality Factor or NF).

The goal of quantification is to generate a signal that is able to represent the characterization of the data in a numeric way. This means that data becomes comparable and able to model. The strategy followed to generate the quantification signal was: assigning a weight to each cluster generated by the AOI algorithm.

Those weights were assigned according to the generalization-level of the attributes at the moment of conforming the cluster. That way, the higher the generalized-level, the higher the ambiguity of the concept, and the lower the weight. The aim was to represent how much representative (numerically) are the clusters at the time to say whether a current working-state can be considered correct or not. At this point, elements with low quantification values (lower than a specified threshold) will be considered as anomalies.

The process of estimation of RCA is a specification or refinement of the Detection results. The RCA estimation process requires the previous definition of variable and failure-types relations (i.e. If an anomaly occurs with high temperatures but low-pressure values, it is possible to have a problem on the component X). Thus, domain knowledge is mandatory if accurate results are expected.

At this point, the RCA result is the description of the clusters of those work-states that can be considered as anomalies (working states not registered in the knowledge-base). The algorithm for RCA calculation inspects the instances within the abnormal dataset (work-cycle) and groups the descriptions of the clusters in order to show them. When clusters are grouped, keywords referred to anomalies are found in the descriptions (those keywords must be defined initially by the domain experts). The output is a list of possible anomaly descriptions referred to the inspected abnormal work-cycle.

For equipment RUL estimation, the previously defined quantification strategy is utilized, based on the ambiguity-weights of the clusters. The idea here is to model the quantification signal in order to estimate when will the next anomaly or breakdown occur.

Within Goizper use case, the RUL estimation process is performed as a combination of the quantification signal obtained by the appliance of AOI algorithm, and Auto Regressive Integrated Moving Average (ARIMA) statistical time series forecasting models. A common objective of Time Series Forecasting methods is to learn from previous data in order to be able to make predictions of future behaviours. The knowledge base generated by AOI algorithm let us to check whether a machine state is already registered in the knowledge base. Furthermore, the order of appearance of machine states can be also evaluated, which provides knowledge about machine behaviours.

Figure 15 shows the different phases explained previously.

To calculate RUL, the establishment of a threshold that represents the minimum value for an element to be considered normal must be defined. That value was defined as the lowest of the NF values registered in the initial cluster

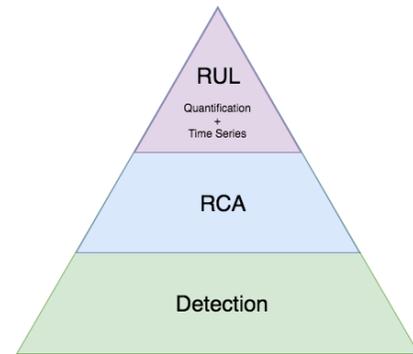


Fig. 15. Different Phases of the development.

generation process. The reason was that all the clusters created are related to a normal behaviour of the machine, so any other clusters with lower weight than the initially created ones could be considered abnormal.

By applying ARIMA time series forecasting models, the Normality Factor evolution is modeled. As a final result, the Normality Factor model allows to predict the wear of the Normality Factor, providing the RUL of the machine in terms of clutch-brake cycles. Finally, clutch-brake cycles are translated into days, by combining the number of cycles the clutch-brake system does per day.

VIII. RESULTS AT PLATFORM LEVEL

The main results obtained after implementing the reference architecture and the data analysis performed over the data collected are:

- A platform that accommodates different industrial processes and assets data for CBM analysis.
- Integrate an interoperable data model for CBM.
- A data/protocol converter that enables translations between most common data formats and protocols.

Regarding data analysis, a small experiment has been performed as a proof of concept in order to show and demonstrate the ability of the proposal. Datasets collected from the clutch-break are organized into breaking datasets and clutching datasets. For this study, breaking datasets were used to perform the analyses.

Many features of the clutch-break machine have been used in the cluster generation step, such as: trigger, angular position, application pressure, line pressure and flywheel speed. First, the clusters were generated by the help of the AOI algorithm, taking into account the order of appearance of the clusters in reference to each instance of a work-cycle. To detect an anomaly on the behaviour, the value of the Normality Factor had been calculated. As told before, that value was established according to the minimum value registered in the cluster-generation step. In this experiment, the value was 0.70, with maximum value of 1.0 and minimum of 0.0 ([0.0, 1.0]).

The representation of the Normality Factor signal shown in the Figure 16 should be the result of calculating the Anomaly Detection step over the training data utilized to generate the

id RCA	RCA Description
1	The variable POSITION has lower values than expected. Possible out of bounds.
2	The variable APPLICATION PRESSURE has higher values than normal. Possible wearing of material X.
3	The SHAFT SPEED is lower than expected. Possible encoder disconnected.

TABLE V
EXAMPLE OF OUTPUT OF AN RCA ANALYSIS.

knowledge base. In this experiment, next two hundred and fifty breaking work-cycles have been predicted. As it can be observed, there are five different work cycles cutting the established Normality Threshold; thus, it can be inferred that five different anomalies are detected.

After the anomalies were detected, the next step was to determine the reason of their occurrence. The RCA algorithm was applied and the results explaining the possible reasons for those anomaly occurrences are expressed in the Table V. Those should be suggestions for the operators to help them to understand the possible reasons of the anomaly or breakdown.

Finally, in order to estimate the RUL for this experiment, the ARIMA Time-Series model was applied over a test dataset in which the machine started working correctly and continue degrading. The predictions made by the ARIMA model with the help of quantification provided by AOI, are shown in Figure 17. The blue line indicate the real degradation of the machine, and the orange line the forecast.

The X axis represents the number of work-cycles in hundreds of cycles, and the Y axis, the probability for the work-cycle to be considered correct.

In order to evaluate the effectiveness of the proposed method we used both historical data comparison and knowledge of the domain experts. As the power of the proposed approach resides on using both monitored data and knowledge of domain experts of the area, it is logical to test the results taking them into consideration.

We had experimental data in which there were data related to normal working cycles of the machine, and many other related to abnormal working cycles of the machine. So, in order to determine the effectiveness of the AD solution, we tested the coincidence between the work cycles in which an anomaly is registered in real life and the predictions of the work-cycles in which an anomaly is detected by the algorithm. The coincidence level is very high, almost 100%. The RUL estimation was tested comparing the number of work-cycles until a failure in real data, and the predicted work-cycles until a failure provided by the algorithm. In some cases in which no failure data was registered in the dataset, it was also estimated following the logic of the results for the domain experts. Finally, the RCA estimation was validated by the domain experts, analyzing the logic of the description of the possible error cause when an anomaly was detected.

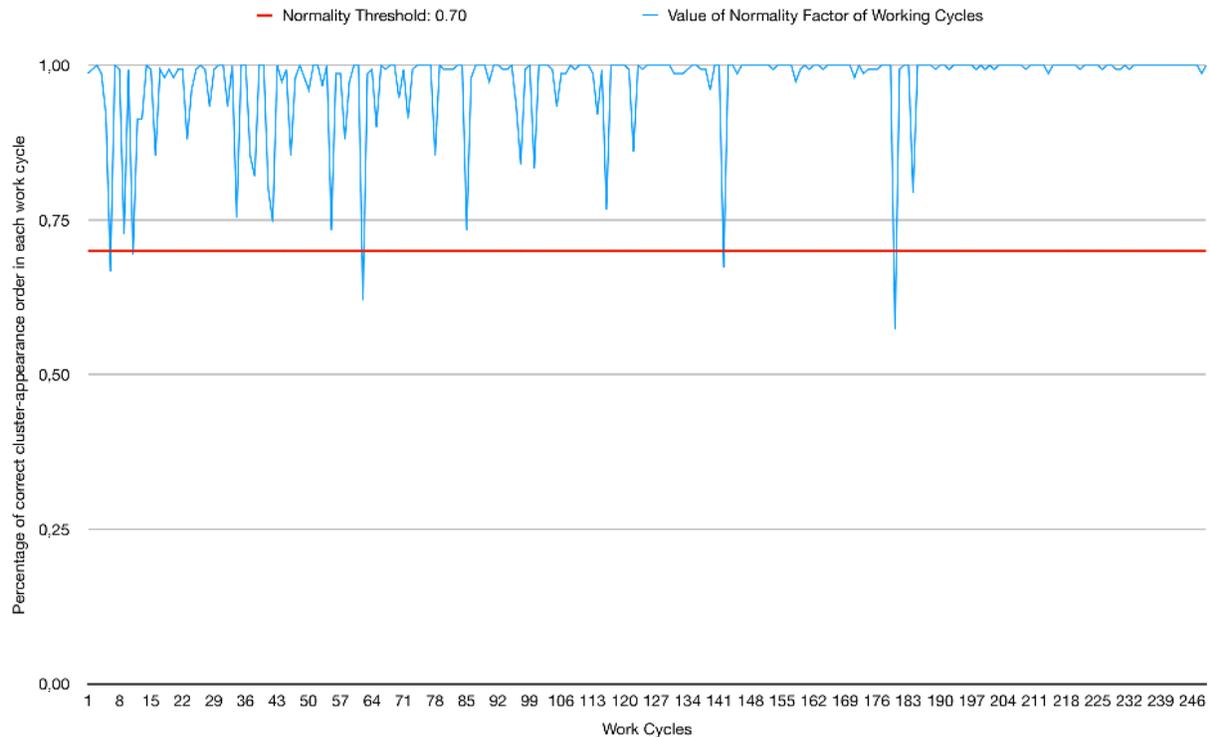


Fig. 16. Graphical representation of Anomaly detection based on the evolution of Normality Factor.

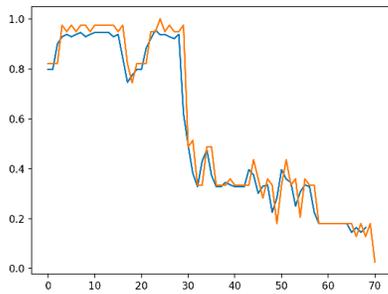


Fig. 17. RUL prediction for a test dataset using ARIMA and AOI.

IX. CONCLUSIONS

This paper describes a specific implementation of the MANTIS reference architecture. The technologies and tools employed to articulate the components for an Industry 4.0 platform are presented. The methodology proposed in the project has been also followed. This methodology focuses on the stakeholder requirements and provides well-structured steps necessary to build the solution. Further, the implementation of the platform enables the provision of maintenance services based on the data collected from sensors and CPS. As other implementations of the same reference architecture [16] [17], the objective is to focus on proactive maintenance services to improve asset availability at lower costs through continuous process and equipment monitoring and data analysis. Attribute Oriented Induction Clustering and Time Series Forecasting are being used to determine Root Cause Analysis powered and Remaining Useful Life. Although only a set of parameters have been analyzed, the solution have been obtained, the solution provides the mechanisms to analyse data and estimate valuable maintenance parameters.

ACKNOWLEDGMENT

This work has been developed by the intelligent systems for industrial systems group supported by the Department of Education, Language policy and Culture of the Basque Government. It has been partially funded by the MANTIS project. The project has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 662189. This Joint Undertaking receives support from the European Union's Horizon 2020 research and Ministerio de Industria, Energía y Turismo (Spain). It has also been partially funded by the Basque Government.

REFERENCES

- [1] Jantunen, E., Zurutuza, U., Albano, M., di Orio, G., Maló, P., and Hegedus, C.: The Way Cyber Physical Systems Will Revolutionise Maintenance, presented at the 30th Conference on Condition Monitoring and Diagnostic Engineering Management, (2017).
- [2] The MANTIS consortium, "The mantis project website," <http://www.mantis-project.eu/>
- [3] <https://www.oasis-open.org/committees/soa-rm/faq.php>
- [4] ISO/IEC, ISO/IEC/IEEE 42010:2011 "Systems and software engineering - Architecture Description", ISO/IEC, 2011.

- [5] C. Hegedus, P. Varga and I. Moldovan "The MANTIS architecture for proactive maintenance" 5th International Conference on Control, Decision and Information Technologies CODIT 2018 Thessaloniki (Greece) (accepted).
- [6] Industrial Internet Consortium, "The Industrial Internet Reference Architecture," Technical Report, 2015.
- [7] K. Pohl, H. Hnninger, R. Achatz and M. Broy, Model-Based Engineering of Embedded Systems: The SPES 2020 Methodology, Springer Publishing Company, 2012.
- [8] <https://www.rabbitmq.com/>
- [9] Vinoski, S. (2006). "Advanced Message Queuing Protocol". IEEE Internet Computing 10: 87-89.
- [10] <https://wso2.com/>
- [11] MIMOSA. Retrieved from: <http://www.mimosa.org/mimosa/>
- [12] ISO 13374-1: Condition monitoring and diagnostics of machines - Part 1: General Guidelines.
- [13] <https://spark.apache.org/>
- [14] Han, J., Cai, Y., and Cercone, N. (1992). Knowledge Discovery in Databases: An Attribute-Oriented Approach. In Proc. 18th Int. Conf. Very Large Data Bases, Vancouver, Canada (pp. 547-559).
- [15] Cheung D.W., Wai-Chee Fu A., Han J. (1994) Knowledge discovery in databases: A rule-based attribute-oriented approach. In: Raś Z.W., Zemankova M. (eds) Methodologies for Intelligent Systems. ISMIS 1994. Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence), vol 869. Springer, Berlin, Heidelberg.
- [16] L. L. Ferreira, M. Albano, J. Silva, D. Martinho, G. Marreiros, G. di Orio, P. Maló, and H. Ferreira, "A pilot for proactive maintenance in industry 4.0," in Factory Communication Systems (WFCS), 2017 IEEE 13th International Workshop on. IEEE, 2017, pp. 1-9.
- [17] Csaba Hegedűs, Paolo Ciancarini, Attila Frankó, Aleš Kancilija, Gregor Papa, Špela Poklukar, Mario Riccardi, Alberto Sillitti and Pal Varga: "Proactive Maintenance of Railway Switches", 5th International Conference on Control, Decision and Information Technologies CODIT 2018 Thessaloniki (Greece) (accepted).
- [18] R. Keith Mobley, "An Introduction to Predictive Maintenance", Elsevier Science (USA), 2002.
- [19] Venkat Venkatasubramanian and Raghunathan Rengaswamy and Kewen Yin and Surya N. Kavuri, "A review of process fault detection and diagnosis: Part I: Quantitative model-based methods", Computers & Chemical Engineering, Volume 27, Issue 3, Pages 293-311, 2003.
- [20] Peter Chemweno and Ido Morag and Mohammad Sheikhalishahi and Liliane Pintelon and Liliane Pintelon Peter Muchiri and J. W. "Development of a novel methodology for root cause analysis and selection of maintenance strategy for a thermal power plant: A data exploration approach", Engineering Failure Analysis 66 (2016) 19–34, 2016.
- [21] Marc Solé & Victor Muntés-Mulero & Annie Ibrahim Rana & Giovanni Estrada, "Survey on Models and Techniques for Root-Cause Analysis", arXiv:1701.08546 [cs.AI], submitted on 2017.
- [22] Lisa Abele and Maja Anic and Tim Gutmann and Jens Folmer and Martin Kleinstueber and Birgit Vogel-Heuser, "Combining Knowledge Modeling and Machine Learning for Alarm Root Cause Analysis", 7th IFAC Conference on Manufacturing Modelling, Management, and Control, 2013.
- [23] Klaus Julisch, "Clustering Intrusion Detection Alarms to Support Root Cause Analysis", ACM Transactions on Information and System Security (TISSEC), Volume 6 Issue 4, pages 443-471, 2003.
- [24] Farzaneh Ahmadzadeh and Jan Lundberg, "Remaining useful life estimation: review", Int J Syst Assur Eng Manag (Oct-Dec 2014) 5(4):461–474, 2014.
- [25] L. Fortuna, S. Graziani, A. Rizzo, and M. G. Xibilia, "Soft sensors for monitoring and control of industrial processes". Springer Science & Business Media, 2007.
- [26] L. Ljung, "System Identification Theory for the User Second Edition". Prentice Hall PTR, 1999.
- [27] E. Jantunen, G. Di Orio, F. Larrinaga, M. Becker, M. Albano, and P. Mal, "A Framework for Maintenance 4.0". 10th IMA International Conference on Modelling in Industrial Maintenance and Reliability (MIMAR). 2018.
- [28] F. Larrinaga, J. Fernández, E. Zugasti, I. Garitano, U. Zurutuza, M. Anasagasti, M. Mondragon, "Implementation of a Reference Architecture for Cyber Physical Systems to support Condition Based Maintenance." 2018 5th International Conference on Control, Decision and Information Technologies (CoDIT), Thessaloniki, 2018, pp. 773-778.