Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 72 (2018) 1339–1344

www.elsevier.com/locate/procedia

51st CIRP Conference on Manufacturing Systems

Validation of PERFoRM reference architecture demonstrating an application of data mining for predicting machine failure

Nandini Chakravorti[a*], M. Mostafizur Rahman[a], Mohamed Redha Sidoumou[a], Nils Weinert[b], Frederik Gosewehr[c], Jeffrey Wermann[c]

[a]The Manufacturing Technology Centre Ltd,Pilot Way, Ansty Park, Coventry, CV7 9JU, UK
[b]Siemens AG, Research in Digitalization and Automation, 81739 Muenchen, Germany
[c]Hochschule Emden/Leer, Constantiaplatz 4, 26723 Emden, Germany

* Nandini Chakravorti. Tel.: +442476647355; *E-mail address:* Nandini.chakravorti@the-mtc.org

Abstract

The PERFoRM project aims to develop a reference architecture for Agile Manufacturing Control systems for plug-and-produce devices, robots and machines. The aim of the work is to improve the flexibility of the mechanical manufacturing of the housing parts involved in the production of industrial compressors and gas separators. An industrial demonstrator has been designed to implement a Data Analytics tool that provides rules beneficial for root cause analysis and a decision support system for early prediction of the failures. The tool also identifies key alarms for monitoring the machine condition.

Peer-review under responsibility of the scientific committee of the 51st CIRP Conference on Manufacturing Systems.

Keywords: Industrie 4.0, Predictive Maintenance, Data Mining, Machine Learning, Condition Monitoring, Equipment health

1. Introduction

The changes in the economic, political and technological landscape in last few decades have had a big effect on manufacturing [1]. Customers demand customised, cheaper and higher quality products [2], whilst manufacturers need to minimise the effect of unplanned equipment down time due to faults. Equipment breakdown can cause long production line stoppages, high maintenance costs and low product quality. Well planned maintenance assists with keeping the equipment in a healthy condition, and helps to decrease the risk of large scale damages of machine and minimise loss of revenue [3].

The Production harmonized Reconfiguration of Flexible Robots and Machinery (PERFoRM) project is focussed on supporting the industrial requirements of enhanced flexibility and configurability [2]. The overall aim of the PERFoRM project is to develop a common reference architecture for Agile Manufacturing Control system for true plug-and-produce devices, robots and machines.

Four diverse use cases, such as the: (a) production of compressors, (b) assembly of low cost full electric vehicles with high variants and high quality on low budget assembly lines, (c) production of aerospace components within high variants and (d) production of consumer white goods have been selected for the validation of the overall architectural principles of the PERFoRM project.

Creating a reconfigurable machining system has caught the attention of researchers from several fields including software development, manufacturing technologies, and networking. This kind of architecture requires collaborations of considerable effort in the above mentioned areas. A reconfigurable architecture can lead to considerable time saving, resources optimisation and adaptability. A number of prior research has report varying levels of success in the application of such as architecture. PROMISE [4] has a targeted the interaction between the three levels of the control pyramid (Enterprise Resource Planner (ERP), manufacturing

Peer-review under responsibility of the scientific committee of the 51st CIRP Conference on Manufacturing Systems.
10.1016/j.procir.2018.03.136

execution systems (MES), and Field). The focus on this approach is at the management level. It includes a meta-model for ontologies and demonstrates the model using agents to cope with changes. If a plant is modified, only a portion of the model needs to be modified to reflect the changes. Low level operations has to be defined separately. The GRACE project [5] reports an intelligent modular distributed manufacturing control systems with a focus on operations on quality. The architecture is linked to an MES system where a multi-agents architecture is applied. The SMARTPRODUCT project [6] focuses on simulating the behaviour of intelligent physical objects in a distributed architecture applied to smart aircraft manufacturing. Intelligent physical objects can assist and interact with designers and workers. The work described on [7] deals with a pluggable architecture of the MADAM system [8] supporting context heterogeneity, wherein functionalities can be plugged and unplugged from/unto the system.

The prior work described above didn't achieve sufficient technical maturity and critical mass to allow large industrial scale uptake of the agile, plug-and-produce concept. The vision of the PERFoRM project is to harmonise the efforts of the past work, and create large scale industrial demonstrations.

1.1. The use case

The use case for the manufacture of compressors for oil and gas applications such as air separation units or for LPG production have been selected for illustration within this paper. The production typically involves small batch sizes down to a single part. The main processes involved are machining, heat treatment, balancing, assembly and painting. A failure or machine breakdown can lead to delays in production or can lead to missing parts to semi-finished products. The objective of this use case is to improve the flexibility of manufacturing, in particular focusing on the mechanical manufacturing of housing parts. Condition based predictive maintenance is considered to be the best approach for improving equipment reliability, reducing production costs due to failure, reducing costs due to maintenance and minimising unplanned downtime. Within the current context, a predictive maintenance system is required to monitor the health of three turning machines. The

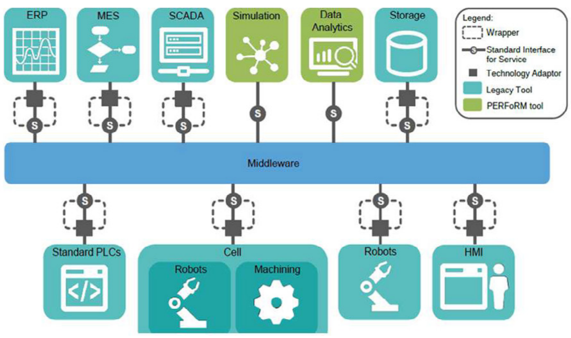

Figure 1: PERFoRM overall architecture

use of a predictive maintenance system can lead to the generation of appropriate maintenance tasks and enable the generation an optimal combination of production and maintenance tasks.

2. PERFoRM Architecture

The aim of the PERFoRM project is to introduce a next generation of agile manufacturing systems that are dynamically reconfigurable to enable self-organisation and adaptation along the system life-cycle. These systems are based on modular plug-and-produce components within the manufacturing system life cycle [9]. The overall system architecture as proposed within the PERFoRM project can be seen in Figure 1, and comprises of a diverse range of hardware and software assets connected via an industrial middleware [10]. The different elements of the proposed architecture are presented in the following sub-sections.

2.1. Middleware

The responsibility of the middleware is to ensure a transparent, secure and reliable communication between the diverse hardware resources (robots, Programmable Logic Controllers (PLCs)) and software applications (e.g. MES and SCADA) representative of a typical manufacturing production facility. The middleware solution used within the work reported in this paper is based on the Apache ServiceMix [11], which is an open-source container from the Apache software family to build a Service-Oriented Architecture (SOA)-based Enterprise Service Bus, such as message broker (Apache ActiveMQ), routing and integration patterns (Apache Camel) and Web service interfaces (Apache CXF) [12].

2.2. Standard Interfaces

One of the key Industrie 4.0 challenge is the interoperability in real industrial environments, requiring the seamless exchange of data originating from a wide range of diverse resources, which may include legacy hardware devices (e.g. robots with specific controllers) and software applications such as databases, logistic tools etc. The PERFoRM architecture proposes to address this challenge via the use of standard interfaces, thus enabling each component to describe and expose its services in a standard way.

2.3. Technological Adaptors

The PERFoRM middleware together with the proper adaptors permit the interconnection of heterogeneous legacy hardware devices, e.g. robots and the respective controllers, and software applications such as databases, SCADA applications and other management, analytics and logistics tools [13].

2.4. Data Analytics and Visualisation

The Data Analytics component uses various sources of machining and process data and involves processes such as data acquisition, pre-processing, analysis and interpretation. This paper focusses, in particular, on the implementation and validation of the Data Analytics component. Further details of the Data Analytics component is present in Section 4.

3. Use-Case Specific Architecture

The overall architecture of the production of industrial compressors use case can be seen in Figure 2. The architecture includes a repair / maintenance management system (MMS) and the production and machine data acquisition system (PDA/MDA) which also includes data from Enterprise Resource Planning (ERP) system. It is envisaged that the machine conditions will be monitored by additional sensors which will automatically publish the data through the middleware (see Figure 2). Additionally, standard interfaces (letter “S” as seen in Figure 2) will be used to enable plug-ability and interoperability.

The Architecture of the Data Analytics tool is presented in Figure 3. The Data Analytics tool will access the data via the

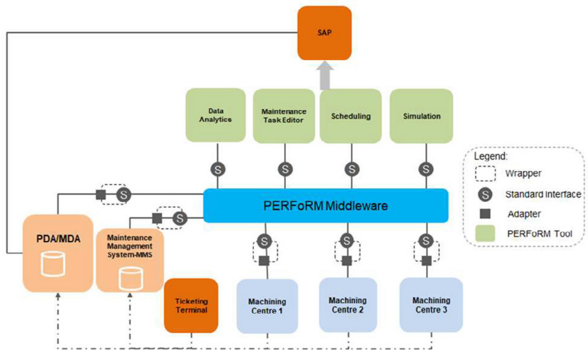

Figure 2: Use Case specific architecture

middleware, analyse it and visualise the result. The results generated by the Data Analytics tool will be used for the creation of new maintenance tasks within the Maintenance Task Editor. The Scheduling tool then accesses the maintenance tasks and proposes schedules for the production and maintenance tasks. After evaluations of the schedules are done by the Simulation tool, the most appropriate maintenance task will be transferred to the SAP ERP system. It is to be noted that this paper reports the functionalities of the Data Analytics tool and also the validation of the PERFoRM reference architecture within the context of the selected use case.

4. Framework for Data Analytics Component

In order to analyse the data obtained from the plant, a framework has been developed (Figure 2). As seen in Figure 2, the data will be transferred from the MMS and the PDA/MDA database over the PERFoRM middleware. The proposed framework also utilises data from appropriate maintenance manuals. The proposed framework includes a Backend System for processing the data and a Front-end enabling the visualisation of data. The Data Analytics tool is responsible for: (1) pre-processing the data from the failure records and maintenance tickets and alarms, (2) aggregating the data and (3) identifying trends within the data. Details of each block within Figure can be seen below:

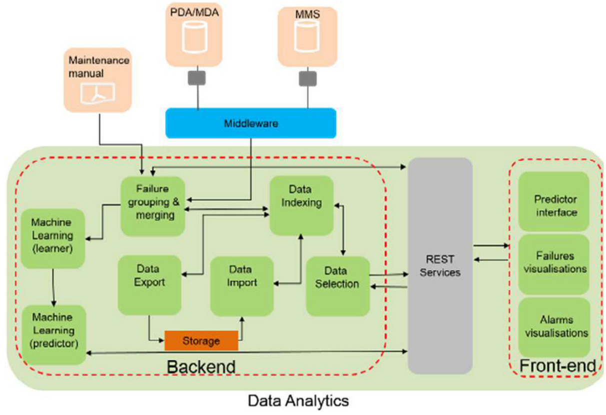

Figure 3: Components of the Data Analytics tool

- Backend System: The Maintenance database contains description of the problem reported in an unstructured textual form. Though this textual form is not the failure type, it may give an indication of the failure type. Consequently, it was necessary to label the information in order to determine the type of the machine failure from the textual form (the reported problem). Information of the prescribed machine failures reported within the Maintenance manual is being used for labelling the failure records, therefore the inputs for the Backend System are the data from the PDA/MDA, MMS systems and the Maintenance manual.

- Failure Grouping & Merging (Labeller): The first step for the pre-processing involves the categorisation of the failure records. An algorithm using Text mining approach has been implemented to label and group the failure records. This approach uses a measure of the distance between the failure description and sections of the maintenance manual. The shorter the distance the more likely the failure description is close to a particular section of the maintenance manual. Sixteen different failure sections (see Table 1, each section represents a type of failure) have been observed in the machine maintenance manual. It is to be noted that the maintenance manual is written in German. TF*IDF [14] have been used to convert the texts associated with a failure and sections of the manual into vectors. TF*IDF is the product of two statistical terms, TF and IDF. TF is the term frequency in a document. IDF is the invert document frequency, i.e. how representative a keyword is for a document (section or failure description in our case). In other words, if a word is present in one section and absent in all other sections then this word has higher chance to representing this particular section/failure description and consequently leads to a higher IDF value. Vectors have been used as they are useful for calculations involving distances. Subsequently, a cosine formula has been used to measure the distance between vectors. The cosine formula is defined as follows:

$$\cos(A, B) = \frac{A.B}{\|A\|_2 \|B\|_2} \quad (1)$$

Where A and B are two vectors representing a failure description and a section of the manual respectively.

It was observed that there were lack of space between the words in the manual and failure descriptions, and this is attributed to be a feature of Germanic languages. The lack of spaces lead to the algorithm not functioning appropriately. Consequently, a de-compounder has been written based on a dictionary to divide long complex words into smaller simpler words. This has been applied exclusively to the failures description as these kind of words appeared frequently in this group. The original data source contained 1700 failure records. This method allowed a labelling of the failure data using the 17 different labels (16 sections of the manual + other type) identified from the manual. An expert based evaluation of the algorithm has been carried out by an operator at the shop floor. The operator labelled around 30 failure based on his knowledge. A comparisons of the labels identified via the manual and automated approaches were conducted. It was observed that the automated approach achieved an accuracy of 60%. Further work is being conducted to investigate ways to enhance the accuracy.

- Data Indexing: Millions of alarms have been recorded in the database systems. An efficient way of querying the database was needed to handle the high volume data in very short time. Once the data are grouped and merged, an indexing operation has been used to process the data using dates. The purpose of indexing is to enable a faster response to queries. Consequently, the selection of the relevant data (depending on the visualisations) can be done in a few memory accesses. In-memory indexing allowed an increase of speed of a magnitude of 10 to 100 times compared to standard database accesses thus allowing real-time visualisations.
- Data Export: This module is used to store the indexed data on disk.
- Data Import: This module loads the saved data from the disk into the memory.
- Data Selection: This module creates a subset of the prerequisite data from memory depending on the requests from the Front-end.
- Front-end: This module is used for visualising the failures and the alarms dataset. The visualisation of failures and alarms will provide a useful insight on the past events and also support the generation of appropriate maintenance tasks. The Front end uses the pre-processed data and creates visualisations (see Figure 4 and Figure 5) for the failures and the key alarms. Figure 4 shows the distribution of different groups (as seen in the Table 1) of a particular machine for a given period of time. Figure 5 illustrates the alarm trends for each failure group for a given period of time. This purpose of this visualisation is to monitor the condition of the machine by analysing the trends of key alarms and potentially predict potential failures.
- Machine learning for predictive model: After pre-processing (labelling and combining the alarms and failure data) and balancing the data, machine learning techniques are applied to generate a predictive model to extract decision rules. Table 2 illustrates some of the sample decision rules. The Decision trees models are commonly used in data mining to explore and classify the data. These decision rules can also be used for conducting root cause analysis of the failures and also for the implementation a decision support system that enables the detection of failures. Identification of key alarms associated to a particular failure can also be conducted. The induced tree and its associated rules will be used to make predictions [15].

Table 1: Failure groups

No	Failure Types in German	Failure Types in English
1	DREHZENTRUM	ROTARY CENTER
2	MASCHINENBETT	MACHINE BED
3	TISCH	TABLE
4	STÄNDER	STANDS
5	VERFAHRBARER QUERBALKEN	PROCESSABLE CROSSBARS
6	SCHLITTEN	SLIMS
7	TELLERMAGAZIN	TELLER MAAGAZINE
8	KETTENMAGAZIN	CHAIN MAAGAZIN
9	ZUBEHÖRTEILE UND WERZEUGHALTER	ACCESSORIES AND TOOL HOLDER
10	HYDRAULIK- UND SCHMIERMITTELKREISLÄUFE	HYDRAULIC AND LUBRICATING CIRCUITS
11	KÜHLANLAGE	COOLING SYSTEM
12	PNEUMATISCHE ANLAGE	PNEUMATIC INSTALLATION
13	SPÄNEFÖRDERER	SPRING CONVEYORS
14	SCHUTZKASTEN	PROTECTION BOXES
15	MOTORGESTEUERTE TASTATUR	ENGINE CONTROLLED KEYBOARD
16	ELEKTRISCHE AUSRÜSTUNG	ELECTRICAL EQUIPMENT
17	ANDERE	OTHER

Table 2. Sample Decision rules extracted from decision tree learning algorithm for prediction of a particular failure

Rules / Alarm patterns
(Rule1) IF Alarm-10860 = No AND Alarm-20050 = No AND Alarm-700333 = No AND Alarm-510308 = Yes AND Alarm-700540 = No THEN failure = Yes

(Rule2) IF Alarm-10860 = No AND Alarm-20050 = No AND Alarm-700333 = No AND Alarm-700543 = No AND Alarm-10208 = Yes AND Alarm-67051 = No AND Alarm-700754 = No: THEN failure = Yes

(Rule3) IF Alarm-10860 = No AND Alarm-20050 = No AND Alarm-700333 = No AND Alarm-700543 = No AND Alarm-601012 = Yes AND Alarm-700636 = No: THEN failure = Yes

5. Test Demonstrator Architecture

A demonstrator has been set up at the Manufacturing Technology Centre (MTC), Coventry, U.K. to test the vertical integration of the Data Analytics tool within the PERFoRM architecture. The architecture for the demonstrator can be seen in Figure 6. The flow of messages within the system i.e. between the databases and the Data Analytics tool can be seen in Figure 6. The purpose of this demonstrator is to validate that the Data Analytics tool can access the maintenance logs and alarms data from the databases via the middleware and can subsequently pre-process and analyse the data. The environmental system requirements of the testing process can be seen below:

- *MySQL database server* hosting the MMS and PDA/MDA databases
- *MySQL database Adaptor*, JDBC-REST interface built on Apache CXF and Spring Boot
- A *middleware* in the form of a preconfigured Apache ServiceMix
- A *Java EE server* (Tomcat 9.0 in this case) hosting the back-end/data analytics tool, the rest services to these tools and the visualisation interfaces

It is to be noted that the maintenance logs and the alarms data have been exported from the MMS and PDA/MDA databases and have been provided to the MTC as CSV files. Due to the unavailability of the original databases within MTC, the contents of these databases have been replicated in to MySQL databases for use within the current context.

For the purpose of the demonstrator, three virtual machines (VM) have been set-up to replicate and enforce the de-coupling between the individual systems. The function of each of the VMs are listed below:

- One VM hosting the database server (MySQL).
- One VM hosting the DB adaptor and the middleware.
- One VM hosting the Data Analytics tool.

The role of the adaptor is to retrieve the values from the database and translate them according to the PERFoRM's data model. In particular, the adaptor performs the following activities:

- Connect to the databases required.
- Execute queries and retrieve the results from the databases connected.

The connection to the database is facilitated by the use of the JDBC (Java Database Connectivity) interface that provides a transparent connectivity to heterogeneous databases. The DB adaptor is currently a REST based web service, which takes care of the conversion of Java objects to and from JSON strings. This can be easily transformed into a web service

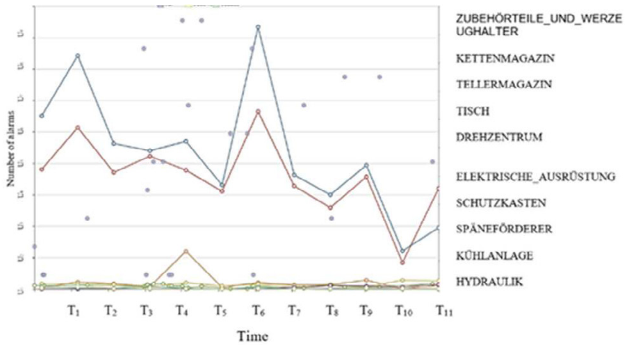

Figure 4: Alarm trends for failure groups per time period, failure groups represented on the right hand-side Y-axis

bundle within the Apache ServiceMix environment using Apache CXF for further integration with the PERFoRM middleware. The creation of the REST based service allows a smooth and flexible integration of the adaptor into the middleware, and facilitates service registration and discovery. The middleware currently facilitates communications by decoupling the DB adaptor with the Data Analytics tool.

6. Validation

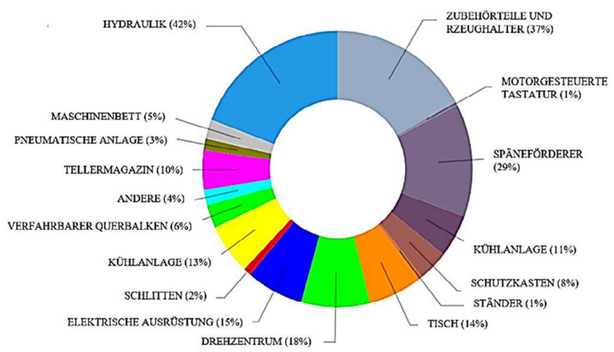

Figure 5. Distribution of failure groups for a given period of time, for a particular machine

In order to validate the proposed architecture, different test cases have been performed. The adaptor and middleware were tested both separately and together for isolation and integration testing respectively. The main driver for normal operation is the Data Analytics tool. The main data flows are (1) to access information from the database and (2) to process this data internally to produce insight. The sequence of operations and data messages for this demonstrator as seen in Figure 6 is as follows:

- Once daily the Data Analytics tool requests the necessary data from the database via the middleware (1)
- Once the middleware receives a request for data from the Data Analytics tool, it sends a request to get the relevant data from the adaptor (2)

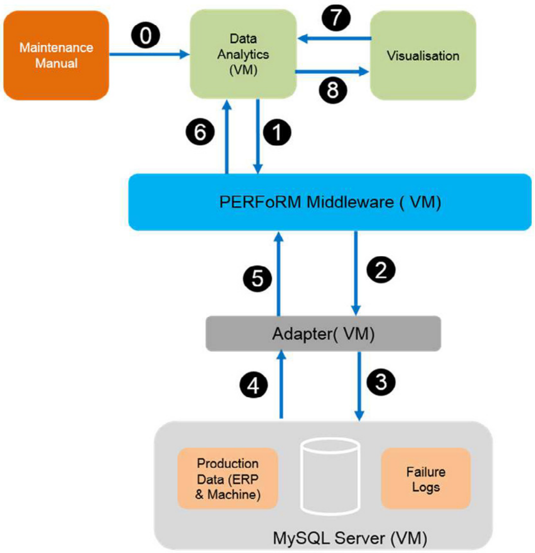

Figure 6: Test Demonstrator architecture and flow of messages

- The adaptor requests the data from the database (3)
- The database delivers the required data to the adaptor (4)
- The adaptor returns the data to the middleware (5)
- The middleware returns the requested data to the data mining tool (6)
- The user requests services through the visualisations which requests the required organised data from the data mining tool (7)
- The datamining tool answers with the organised data for visualisation (8)

The test cases were run with the intention of validating the Data Analytics tool and its integration within the overall PERFoRM architecture. The adaptor and service interface performed as expected. It was observed that the Data Analytics tool was able to access the data automatically through the architecture and services layers described above, for full vertical integration within the PERFoRM architecture was validated. Test visualisations provided by the data analytics tool were also tested to ensure the validity of the end to end testing of the system.

7. Conclusions

A demonstrator has been set up at the Manufacturing Technology Centre (MTC), Coventry, U.K. The objective of the MTC's demonstrator is to test the integration of a Data Analytics tool within the PERFoRM architectural framework. This work has implemented a system that will enable the machine maintenance team to schedule appropriate maintenance well in advance to avoid any unexpected downtime. It is envisaged that the PERFoRM framework (including the Data Analytics tool) will be migrated to the actual factory as a part of the PERFoRM project.

Acknowledgements

The PERFoRM project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 68043.

References

[1] Mourtzis, Dimitris, and Michael Doukas. "Design and planning of manufacturing networks for mass customisation and personalisation: challenges and outlook." Procedia CIRP 19, 2014: 1-13.

[2] Tseng, Mitchell M., and S. Jack Hu. "Mass customization." In CIRP Encyclopedia of Production Engineering, pp. 836-843. Springer Berlin Heidelberg, 2014.

[3] Efthymiou, K., Papakostas, N., Mourtzis, D. and Chryssolouris, G. On a Predictive Maintenance Platform for Production Systems, 2017.

[4] Ferrarini, L., Veber, C., Luder, A., Peschke, J., Kalogeras, A., Gialelis, J., et al. (2006). Control Architecture for Reconfigurable Manufacturing Systems: the PABADIS'PROMISE approach. In IEEE Conference on Emerging Technologies and Factory Automation, 2006. ETFA '06 (pp. 545–552).

[5] Matthias, F., Jäger, T., Turrin, C., Petrali, P., Pagani, A., & Leitao, P. (2013). Implementation of a Methodology for Consideration of Product Quality within Discrete Manufacturing. In B. Natalia (Ed.), Manufacturing Modelling, Management, and Control (Vol. 7, pp. 863–868). Presented at the Manufacturing Modelling, Management, and Control, St. Petersburg,

[6] Miche, M., Baumann, K., Golenzer, J., & Brogle, M. (2012). A Simulation Model for Evaluating Distributed Storage Services for Smart Product Systems. In A. Puiatti & T. Gu (Eds.), Mobile and Ubiquitous Systems: Computing, Networking, and Services (pp. 162–173). Springer Berlin Heidelberg.

[7]] N. Paspallis, R. Rouvoy, P. Barone, G. Papadopoulos, F. Eliassen, A. Mamelli, A pluggable and reconfigurable architecture for a contextaware enabling middleware system, in: Proceedings of the OTM 2008 Confederated International Conferences, 2008, pp. 553–570.

[8] Rouvoy, R., Eliassen, F., Floch, J., Hallsteinsen, S., Stav, E.: Composing Components and Services using a Planning-based Adaptation Middleware. (2008) Lecture Notes in Computer Science, vol 4954. Springer, Berlin, Heidelberg

[9] Leitão, Paulo, José Barbosa, Matthias Foehr, Ambra Calà, Pietro Perlo, Gregorio Iuzzolino, Pierluigi Petrali, Johan Vallhagen, and Armando W. Colombo. "Instantiating the PERFoRM System Architecture for Industrial Case Studies." In Service Orientation in Holonic and Multi-Agent Manufacturing, pp. 359-372. Springer, Cham, 2017.

[10] Gosewehr, Frederik, Jeffrey Wermann, and Armando Walter Colombo. "Assessment of industrial middleware technologies for the PERFoRM project." In Industrial Electronics Society, IECON 2016-42nd Annual Conference of the IEEE, pp. 5699-5704. IEEE, 2016.

[11] Badura, Adam, Bartosz Sakowicz, and Dariusz Makowski. "Integration of management protocols based on Apache ServiceMix JBI platform." In CAD Systems in Microelectronics, 2009. pp. 381-384. IEEE, 2009.

[12] Kwon, Young-Woo, Eli Tilevich, and William R. Cook. "An assessment of middleware platforms for accessing remote services." In Services Computing (SCC), 2010 IEEE International Conference on, pp. 482-489. IEEE, 2010.

[13] N. Chakravorti, E. Dimanidou, G. Angione, J. Wermann and F. Gosewehr, "Validation of PERFoRM reference architecture demonstrating an automatic robot reconfiguration application," 2017 IEEE 15th International Conference on Industrial Informatics (INDIN), Emden, 2017, pp. 1167-1172.

[15] Bastos, Pedro, Isabel Lopes, and L. C. M. Pires. "Application of data mining in a maintenance system for failure prediction." Safety, Reliability and Risk Analysis: Beyond the Horizon: 22nd European Safety and Reliability 1 (2014): 933-940.

[14] Salton, Gerard, and Christopher Buckley. "Term-weighting approaches in automatic text retrieval." Information processing & management 24, no. 5 (1988): 513-523.