I. Aldalur et al., "Advantages of Arrowhead Framework for the Machine Tooling Industry," IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society, 2020, pp. 4511-4518.

DOI: https://doi.org/10.1109/IECON43393.2020.9254868

Advantages of Arrowhead Framework for the Machine Tooling Industry

1st Iñigo Aldalur
Computer Science Department
Mondragon University
Arrasate-Mondragon, Spain
ialdalur@mondragon.edu

2nd Miren Illarramendi
Computer Science Department
Mondragon University
Arrasate-Mondragon, Spain
millarramendi@mondragon.edu

3nd Felix Larrinaga
Computer Science Department
Mondragon University
Arrasate-Mondragon, Spain
flarrinaga@mondragon.edu

4th Txema Perez
Computer Science Department
Mondragon University
Arrasate-Mondragon, Spain
txperez@mondragon.edu

5th Fernando Sáenz
Savvy
Donostia - San Sebastián, Spain
fsaenz@savvyds.com

6th Gorka Unamuno
Ideko S.Coop
Elgoibar, Spain
gunamuno@ideko.es

7th Inaxio Lazkanoiturburu
Ulma Embedded Solutions S.Coop
Oñati, Spain
ilazkanoiturburu@ulmaembedded.com

***Abstract*—Immersed in the digital era and fully experiencing the changes introduced by the new industrial revolution of the so-called Industry 4.0, there are still many aspects of industrial digitization to resolve. Interoperability among devices and machines is one of the challenges. Sensors, components and machines from different vendors work as independent silos offering large amounts of heterogeneous data which relational capabilities are not fully exploited. Quick development, deployment and testing of new software solutions that take advantage of those data is another important matter. The requirements in terms of equipment resources and engineering efforts is high when planning new implementations. Platforms that enable the efficient application of those solutions at the right level (machine, edge, plant or cloud) are also necessary.**

This paper presents an industrial case study on the application of the Arrowhead framework. The framework is implemented in the Machine Tooling ecosystem and enables the integration of grinding machines with other sensors, components or machines. Different software engineering tools offered with Arrowhead are used to design new solutions in Cyber-Physical System and Internet of Things in Industry 4.0 and make them Arrowhead compliant, for fast deployment of platforms and applications (Dockers) or for testing purposes (Management tool). Finally, the potential of agile construction of new applications is analysed by providing an Human-Machine Interface at machine level and the provision of services for data consumption at cloud level.

***Index Terms*—Cyber-Physical System, Internet of Things,Industry 4.0, Arrowhead**

I. Introduction

Advances in computing and communication are leading to the digitalization of industry. Embedded devices in the form of Cyber-Physical Systems (CPS) enable the integration of Operational Technology produced by industrial machinery in the Information Technology world. That is, traditional functionality and data managed by Supervisory Control And Data Acquisitions, Programmable Logic Controllers or similar devices is now available to software applications, data analysis techniques or web monitoring solutions available in Internet. CPS is defined as a system which involves collaboration between two classes of resources: software (cyber) entities and physical devices (which interact, interface or integrate with other physical devices or with the cyber components) [1] The combination of Information and Communications Technology (ICT) and Cyber Physical Systems-of-Systems (CPSoS) is transforming the value chain of manufacturing processes. This transformation ranges from the pure manufacture and sale of physical products to the provision of integrated solutions in which physical products are enhanced by functions and services. This trend is called ”servitization”, i.e. the process of creating value in products by adding services. Servitization can be seen as a form of link channel [2]. Servitization can be ”delivered” at two levels; at plant level or in the cloud. At plant level, the concept aims to extend the paradigm of data computation and service delivery to the local or machine environment for those applications that have strict real time requirements. Recently an architectural concept called ”fog/edge computing” has been introduced [3]. This concept consists of taking intelligence, data analytics and knowledge generation to smaller clouds near physical devices.

The combination of cloud and plant paradigms (fog/edge computing) through CPSoS is allowing companies to evolve their hierarchical and static configuration towards a new architecture characterized by agility and interactions between systems. During the production phase, CPSoS facilitate the integration of data within the company. In the pre-production and post-production stages, cloud-based CPSoS can provide relevant data that can be used to support both Product Lifecycle Management (PLM) and Service Lifecycle Management (SLM) [4].

In this landscape, Internet of Things (IoT) and CPS-based platforms are increasing in their size and target applications in a steady manner. This raises higher level issues that might hamper their implementation in the industrial context: 1) How do devices, components, CPS and platforms integrate and interoperate in this service oriented context 2) Which is the right level (machine, edge or cloud) to interexchange the data produced by those elements? 3) How can we enable the fast development and deployment of new applications based on those data? 4) How can we support engineers and companies with tools and methodologies to develop better software during the whole life cycle of their products/services?

To address these issues the ECSEL Project Productive4.0[1] was born in 2017. The pillars of the project are Digital Production and PLM and the main objective is to achieve significant improvement in digitalizing the European industry by means of electronics and ICT. Productive4.0 relies on the Arrowhead platform[2] to provide interoperability and integration between heterogeneous networks for Industrial IoT. Among the nine product related use cases, Mondragon Corporation proposes the demonstration of Productive4.0 technologies within the machine-tool building sector, so that innovative business models following a product-service scheme will be enabled.

This paper presents the machine tool digitalization use case proposed by Danobat, Savvy, Ideko Ulma, Mondragon Corporation and Mondragon University (MU). First, Section II analyses the problem we want to solve, its causes and its consequences. Section III present the related work and Section IV introduces the use case and the Arrowhead framework. Section V discusses the implementation of the use case. Finally, conclusions and future work are presented in Section VI.

II. Problem analysis

The problem

The great potential and enormous expectations around the IoT and CPS solutions are resulting in challenges that the research community has to address in order to boost the progress and deployment of these solutions in real application contexts [5]. The problems to achieve this aim are the inability to determine relevant data, incapacity to capture and transmit adequately big volumes of data and non-existent relational capacities and trust-building between user-producer to make data accessible [6].

Causes

The main reasons why these problems happen are:

- **Interoperability**: Machines work as independent silos. Communication between machines and applications is vendor controlled not enabling communication with other machines or components. The huge variety and heterogeneous industrial systems makes extremely complex the integration on cyber information technologies [7]. This lack of interoperability hampers the possibility to cross/relate data from different elements to support improvements at machine and at process level.
- **Software and Hardware Engineering requirements**: New software solutions (algorithms, Human-Machine Interface (HMI)) based on plant data require additional equipment and costly integration processes [8]. Sensors and other machines and components to be connected require significant engineering effort [9]. That is, it is difficulty to quickly deploy new solutions due to integration process costs and equipment requirements.
- **Determine relevant data**: CPS connect the cyberworld with various ICTs, based on IoT, cloud, and big data analysis, which is a very complex task [10]. CPS will consistently produce a large amount of heterogeneous data without a clear meaning at different levels (sensor/CPS, machine, plant, cloud level). Companies need to determine relevant data and use special techniques to process this data at the right level. This is the reason why they must provide the necessary equipment to capture, manage and transmit large amounts of data in order to enhance system scalability, security, and efficiency [11] and they must also define strategies for the definition of advance services to apply algorithms, data analysis or other solutions based on data properly [12].

Consequences

The principal effect of the problems detected is:

- **Difficulties for adding new equipment (scalability)**: scalability is the ability of increasing the capacity of the software or hardware service by expanding the quantity of the software or hardware provided [13]. Adding new software and hardware is a must for companies but they are costly in equipment and engineering efforts. This hinders the creation of new development environments where third party applications can be deployed. If the causes of the problem are not mitigated, expanding the CPS will produce a more significant issue.

III. Related work

In recent years, Industrial Internet of Things (IIoT) technology has been commonly utilized in manufacturing systems. Hundreds of initiatives around Digital Manufacturing Platforms and IoT have been carried out in distinct Research and Innovation actions in the European Community (MANTIS, C2Net, CREMA, FIWARE for Industry FITMAN ...). Nonetheless, these platforms have not led to a successful and efficient digitisation of all aspects and resources of manufacturing industry [4]. In fact, German and Chinese industries have been updating their factories in the recent years in order to become more competitive, innovative, and efficient [14]. In this sense, a huge variety of IoT applications have been developed to solve problems in industries [15]. There are also commercial IoT platforms being used in industry such as Amazon's AWS IoT, Microsoft's Azure IoT Suite, IBM's Watson IoT or MathWorks' ThingSpeak. These solutions are closed solutions based on specific technologies making interoperability between different solutions difficult. This is the reason why, the Industrial Internet Consortium has defined the concept of Connectivity Framework: Industrial Internet Connectivity Framework. This platform involves a large number of companies from all around the World and enables the communication between different IIoT systems and components developed by different participants of the ecosystem based on syntactic interoperability to support designers understanding

[1]https://productive40.eu/
[2]https://www.arrowhead.eu/arrowheadtools

standards and choosing the right ones for their applications [16]. In order to support all this, a strong community behind the IoT applications is desired. The community must provide the guidelines, code, tools and support necessary to implement and integrate new solutions using a given framework.

Different works have been presented related with IoT in the last decade. For example, [17] presents a layer structure to represent the IoT technological stack where interoperability is key. The paper also identifies an important gap. They mention that most solutions assume that the personal networks, made of devices and interconnected directly or via gateways, are managed and controlled remotely by Cloud services (typically hosted by the vendor). Standardized reference architectures, instead, do not explicitly mention the cloud backed solutions, leaving space to alternatives, where hosting the IoT applications and IoT service platform is performed in local servers. That is, leaving space for local clouds that could be managed at plant or machine level. [18] sump ups previous works about IoT in a survey referencing architectures, security and interoperability. The paper compares the different reference architectures proposed in scientific papers, existing standards and white-papers published by main vendors (Microsoft, SAP, Intel, WS2O). Their conclusion is that not all the interoperability challenges are currently solved and that research is still necessary. [1] introduces a novel IoT approach based on CPS framework for micro devices. Their IoT framework is the first that processes the domain involving the assembly of micron-sized parts and it explores the feasibility of Software Defined Networking principles to support distributed collaborations involving cyber and physical components within this framework. In this work, the relation between IoT and CPS is close. The reason behind this is that the frontiers between IoT and CPS are not always well defined [19]. CPS and IoT [20] have become extremely important for industry and academia during last years. The fact that they have revolutionised human life since their appearance has withdrawn the interest in the capabilities of these technologies.

Regarding CPS and IoT works, [21] demonstrates the usefulness and scalability of their solution by applying it to real-world CPS. Their solution has been implemented in a plant automation software system. They extract different types of limitations they detect in the plant and they present the mined constraint candidates to users and offers filtering and ranking strategies. [22] presents a practical manner to construct a cross-layer security game model for IoT analysing the CPS quantitative vulnerability and unifying time-based payoff quantification. Another example is [23] that introduces an IoT framework that discovers any physical interactions and generates interaction chains across applications in the CPS environment. The fast development and deployment of new software solutions in the industrial contexts needs to be managed by techniques and tools that support software engineers during the whole software life cycle. There are numerous reference of the importance of proper software engineering methods from documentation [24], development [25], testing [26], continuous integration [27] and deployment [28]. In recent years, the impact of virtualization and dockerization at different levels has drawn the interest of the research community [29] [30] and also for IoT and CPS. "Docker is a tool that promises to easily encapsulate the process of creating a distributable artifact for any application, deploying it at scale into any environment, and streamlining the workflow and responsiveness of agile software organizations" [31]. Docker has been used in IoT such as [29] that proposes a modular and scalable architecture for IoT based on Docker to simplify management and enable distributed deployments. Some other have used Docker technology for CPS such as [30] that defines a CPS middleware for communication among multiple networked modules, whereas the Docker is proposed to wrap up software module. In the next section, we present our solution that diminishes common causes and consequences of the main problem in current CPS and IoT explained in section II.

IV. Use Case Arrowhead framework for Machine tooling industry

To address the issues explained in section II, Mondragon Corporation has proposed a set guidelines and a methodology for the development of advance services in industrial contexts. Figure 1 shows the principal parts of the Servitization Tool. Mondragon Corporation is the embodiment of the cooperative movement and forms the 1st industrial group in the Basque Country and the 7th in Spain. This servitization tool is the digitalization strategy proposed by Mondragon Corporation to the cooperatives in the industrial group. The strategy to design advance services is summarised in the figure.

Danobat Group is a machine tool provider belonging to Mondragon's industrial group that has applied this strategy with the support of Ideko and Savvy. Ideko is a technology centre specialise in advance manufacturing and Savvy is a technology provider in the scope of advance monitoring and services.

With the objective of improving machine tool performance and optimizing manufacturing resources Danobat Group, Ideko and Savvy have joined the Productive4.0 European project. In the use case, the partners aim is to develop smart services on top of CPS, integrate third party applications and select a platform that enables this integration at different levels (machine, plant or cloud). Furthermore, the usage of production data, machine stochastic models and/or simulation techniques upon those smart services will enable the prediction of unexpected machine behaviours and therefore make production more efficient and predictable.

Ulma and MU collaborate in the project as third party sensor and CPS provider and as platform integrator respectively. Ulma provides engineering services along electronic product development life cycle focusing in the development of sensors and CPS for industry. MU is a university with experience in interoperability platforms.

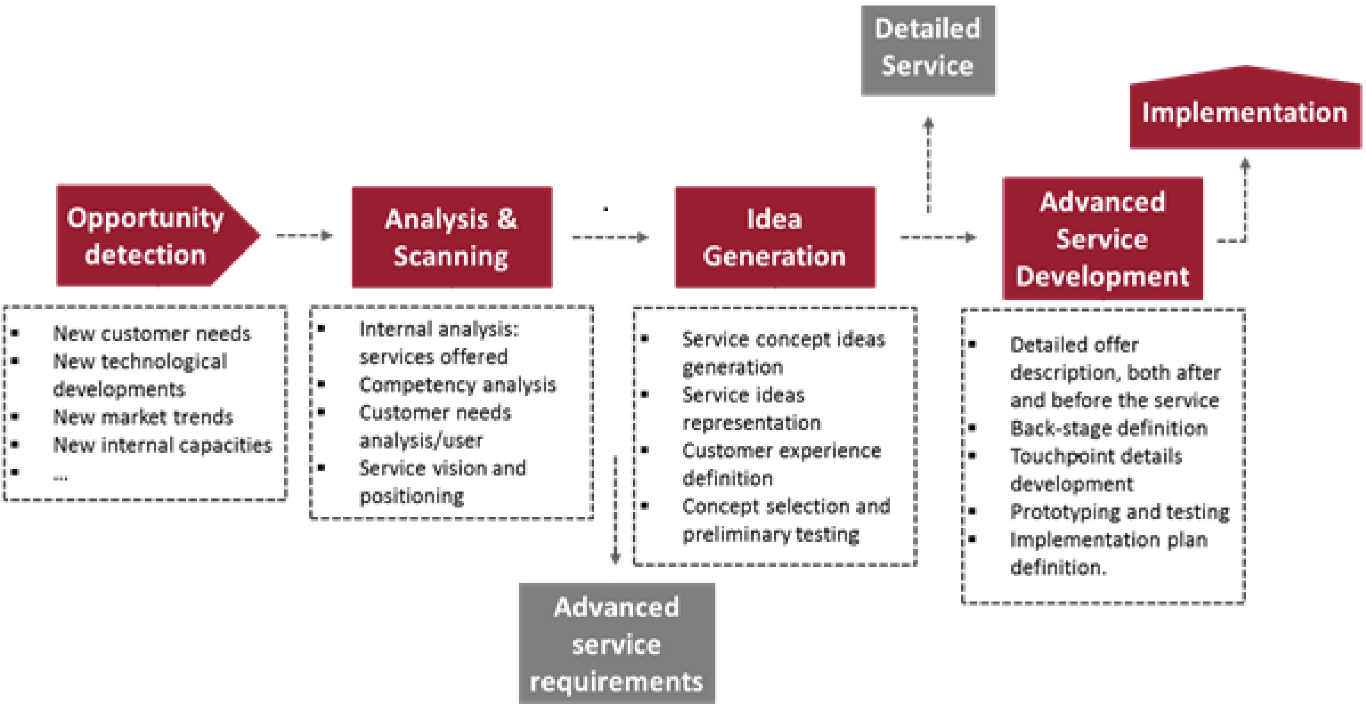

Fig. 1. Servitization Tool for Advanced Services

A. *Objective of the use case*

The main objective of the case study is to move from a business model based only on product (Grinding Machine LG of Danobat), to a model based on the concept PRODUCT-SERVICE where the machine tool offers information and value-added services as support for maintenance, production or quality. This exchange of information is conducted through the Arrowhead framework that offers a systems/services architecture that enables the digitization and automation of production.

This general objective is separated into two specific objectives in the Productive 4.0 project:

- Design and development of an advanced maintenance service at machine-level to improve availability. The machine tool is transformed into a CPS that provides a layer of Arrowhead compliant services and displays its status and condition in real time. In addition, the machine is able to integrate/cooperate with any other Arrowhead compliant system to provide value-added services. If the production system where the machine tool is integrated lacks of a local cloud managed by the Arrowhead framework, the machine tool will deploy Arrowhead core services using Docker containers.
- Design and development of an efficiency service to increase productivity and ensure quality. The production system associated with the machine tool provides complete information on the state of the machine and the manufacturing process. This service combines machine-level and platform-level services.

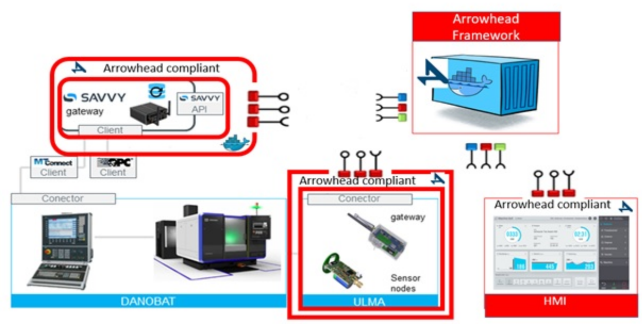

Fig. 2. Architecture of the solution

B. *Architecture and implementation of the Use Case*

In the case study, the Danobat grinding machine provides information on its components, production and context (temperature, vibration, speed, downtime...). Savvy is in charge of activating these functions using a CPS, the SAVVYBOX gateway, which makes possible to exchange data and commands with the platform in the cloud. In addition, using Docker technology, the Arrowhead framework is deployed in the gateway, enabling the interoperability of services at plant level. The architecture of the solution is shown in figure 2.

The architecture of the proposed system allows different modes of customer-supplier relationship ranging from the sale of a high-performance product to the sale of services related to the product that has been acquired. In addition, it is

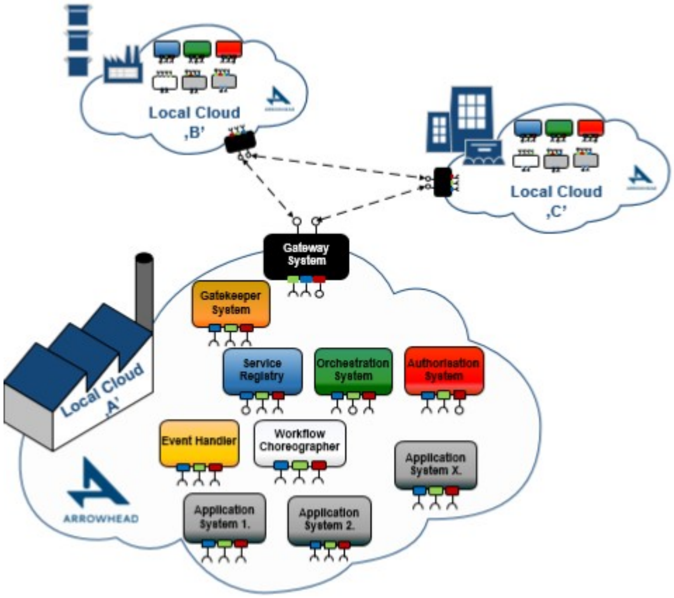

Fig. 3. Arrowhead Local Cloud Architecture

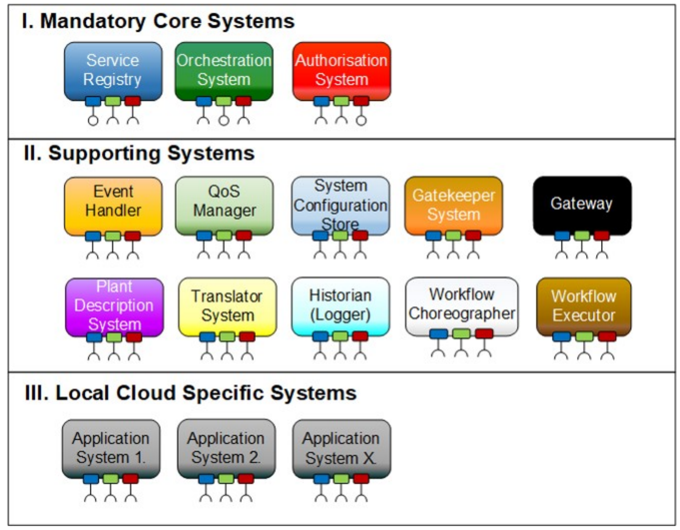

Fig. 4. Arrowhead Types of Systems

easily adaptable allowing the integration of new components (sensors/actuators) and enabling interoperability with other systems (integration of components and/or own integration into the company's management systems).

This architecture is based on the Arrowhead framework and its ability to manage local clouds. A local cloud, as shown in figure 3 , is a service-oriented architecture in a bounded and autonomous environment. It does not need external resources to be formed and contains and manages its own services. The architecture is empowered by inter-cloud communication capabilities connecting different clouds with each other. The Arrowhead framework defines three types of the systems (figure 4). 1) Mandatory Core Systems which provide the necessary functionalities of Service-oriented architecture (SOA) (Orchestration for service discovery and late binding, Service Registry and Authorisation to provide authorisation, authentication and accounting). 2) Supporting Systems that allow general services that are often needed in System-of-System (SoS), such as the Gateway and Gatekeeper Systems for inter-cloud communication, the Workflow Choreographer and Executor for business process execution, the Event Handler to circulate status and event information and the Plant Description System to keep track of SoS or Plant related meta-data. 3) Local Cloud Specific Application Systems which are local cloud-specific applications. These are mostly the local systems; from the smallest sensors up to the biggest CPS.

The use of local clouds allows to:

- Decrease latency times and respond problems in real time.
- Facilitate scalability by enabling the integration of very large systems.
- Enable the integration of new components.
- Increase the security and reliability of the systems.
- Ease engineering work.

In addition, the use of Dockers for the deployment of Arrowhead services enables scalability in a simple and effective way. Starting from a base image it is trivial to launch more service instances. Moreover, the use of Dockers allows isolating Arrowhead services, separating them from other services that may be running on the machine and providing higher levels of security.

Thus, new functionalities can be presented at plant level (fog-edge level), such as the incorporation of a HMI that consume the data provided by the grinding machine or the integration of the grinding machine in plant processes controlled by Manufacturing Execution System, PLM or Enterprise Resource Planning systems or services deployed at cloud level. This feature has been implemented in the use case and it is shown in figure 5. In the SAVVY gateway there is a Arrowhead Local cloud in which services are deployed enabling to monitor and control machine information at plant level. At cloud level, another Arrowhead Cloud is deployed and integrated with SAVVY Cloud Analytics solution offering services at cloud level. These services are related to Smart Maintenance and Production Efficiency.

To demonstrate the connection of other systems/services in the plant, a CPS provided by Ulma is integrated into the solution through Arrowhead's framework. This CPS integrates sensors into the spindles that provide pressure and temperature information.

Data from Danobat's machine and Ulma's CPS, as well as from third parties, can be uploaded to the cloud platform. This cloud platform collects and generates the data needed to provide information to stakeholders and create new tools and services. Third parties can access all this information through applications and services available in the cloud. Thus, innovative business models can be offered (preventive maintenance, life cycle management, etc.).

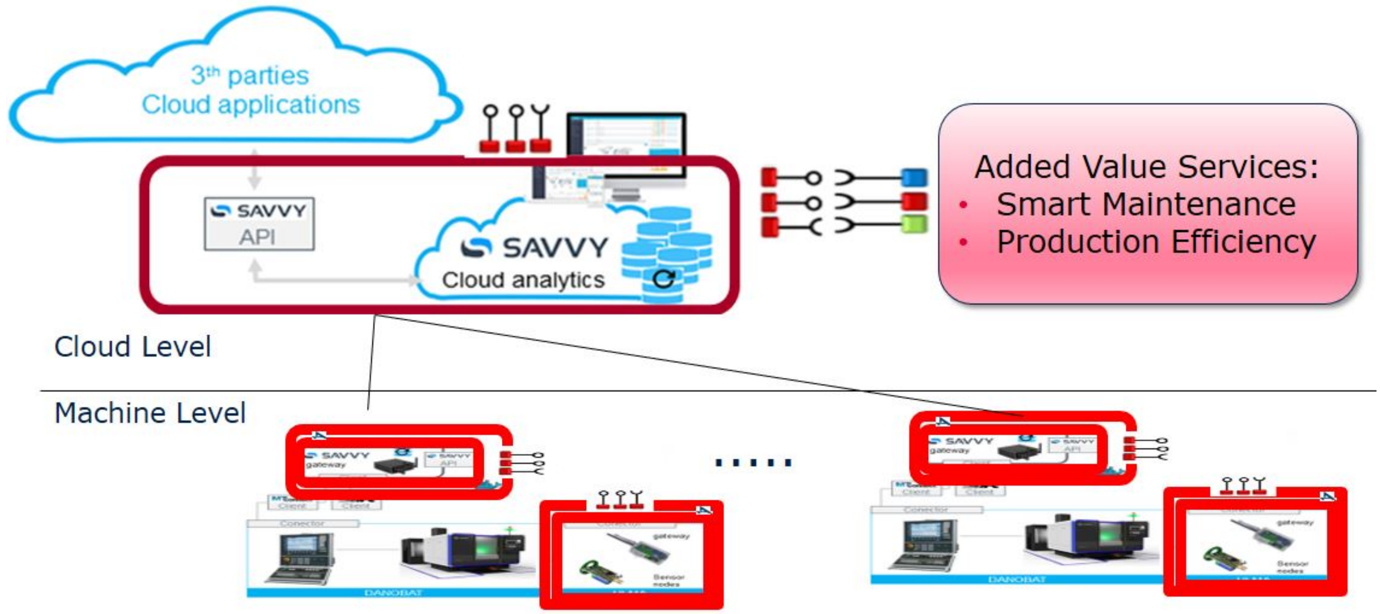

Fig. 5. Arrowhead Local Cloud: Factory level

V. Evaluation

Our approach has been implemented in Ideko demonstrating that this method is real and hence, it could be implemented in different companies.The main results obtained after implementing the Arrowhead framework over CPS are; 1) A platform that accommodates different industrial components and devices enabling their integration and interoperability, and 2) A solution that enables the collection and use of data at the right level. [30] evaluates its approach demonstrating that CPS have been implemented in a real context. In that sense our approach is similar but we additionally want to present the advantages of applying the technologies and frameworks used in the case study and show how they mitigate the causes of the problem identified in Section 2. Based on [21] evaluation, we want to evaluate our solution with the following research questions:

- RQ 1. How useful is Arrowhead for interoperability in CPS?
- RQ 2.What is the effectiveness or saving that we can get in the deployment of a software solution using Dockers?
- RQ 3. What is the scalability level that we can obtain by using Arrowhead at different levels of the architecture?

A. Interoperability

With regard to the first research question, the Arrowhead framework is consolidating as a sound alternative to provide interoperability and integration between heterogeneous networks for IIoT. Several European projects (Arrowhead, Mantis, Productive4.0...) and over 100 companies throughout Europe corroborate the usage of the platform. The use cases implemented in those projects have shown how Arrowhead supports the collaboration of newly built and legacy CPS architectures based on the principles of SOA applying a SoS approach. Arrowhead realises the local cloud concept empowered by inter-cloud communication capabilities. In our use case we have been able to easily integrate a third party provider sensor (Ulma) into the CPS that monitors and manages Danobat's grinding machine using the Arrowhead framework.

We have extensively used the documentation, code examples[3] and tools to deploy the Arrowhead applications for our use case. We have evaluated Arrowhead according to the criteria assessment presented in [32]. Answering the questionnaire presented there, we find Arrowhead highly valuable for understandability, documentation, buildability, installability, learnability, testability, portability and above all interoperability. Additionally the commitment to continue improving the platform from the Arrowhead community give us confidence in terms of governance and supportability.

B. Software and Hardware Engineering Requirements

As for research question two, in the last implementation of the solution we have used Docker technology. The reasons why we have implemented our solution with Docker is that some Docker advantages are appropriate for our use case. 1) Rapid application deployment, Docker includes the minimal run-time requirements of the application, reducing their size and allowing them to be deployed quickly 2) Docker reduces effort and risk of problems with application dependencies 3) Docker images are very small, which reduces the time to deploy new application containers and 4) Docker tracks successive versions of a container, inspect differences, or roll-back to previous versions reusing components from the

[3]https://github.com/arrowhead-f

preceding layers, which makes them noticeably lightweight. Thus, the deployment of the solution is carried out much more easier and the dependencies each specific project have are avoided. In order to measure this, the solution implemented by Danobat is a generic architecture reusable in different projects. Furthermore, the addition of new nodes and services is always done in the same way and the scalability issue topic is solved based on Docker technology.

Further we have been able to offer monitoring capabilities at machine/plant level integrating an HMI that combines and monitors information from different systems/machines and sensors (spindle sensor). The HMI has been implemented also with Docker and it shows data stored in the local cloud uploaded by sensors of Danobat systems.

C. Scalability

Regarding research question three, Arrowhead, software patterns and Dockers enable the easy development, adaptation and integration of new and third party software into the common framework. Docker is independent from the host version and due to this, it is easy to add this software in different new machines, Docker containers can be transferred to another machine that runs Docker, and executed there without compatibility issues. New or legacy solutions based on services can be easily integrated in a operative local cloud enhancing the capabilities to collect and manage data in a given context. Furthermore, Arrowhead support systems also contribute to scalability since they liberate developers from the task of building standard or common services to many applications such as the Event Handler, the Gateway, the Workflow Choreographer or the Data Manager. Our solution has been developed thinking that in the future, more machines would be added in the plant and that it must support this growth without harming the initial implementation.

VI. Conclusions and Future Work

The implementation of the Arrowhead framework in a real machine tooling context to support the servitization of the machine has been a successful experiment where several challenges have been overcome: 1)We have been able to integrate and make interoperable heterogeneous devices, components, CPS and platforms in a service oriented context. 2)We have implemented a framework to support the interexchange of data at the right level (machine, edge or cloud) 3) By using Arrowhead, we have been able to easily develop and deploy new applications based on the data produced in CPS 4) The documentation, code, examples and tools provided by Arrowhead have supported the development of better software in many stages of the services life cycle. A number of lessons have been learned:

- Interoperabiliy issues have been solved. Arrowhead methods and services will also help in our future implementations. A strong community is working in the same direction and we will follow the future research lines of the framework.
- Scalability issue has been improve by Dockers technology. We have to add that this technology needs a learning process and time but once this is done the time needed and efforts to be used in future projects is decreased. Deployments and new architecture designs will be easier to implement in future projects. It is a extremely agile and flexible technology to be used.

Regarding the future lines, the local cloud architecture propose by Arrowhead and the usage of CPS allowed our use case to find the proper level for data management. Now we are able to monitor specific data not only at cloud level but also at machine/plant level. Only specific services and data from this cloud will be available to other local clouds. In our use case, the information and services from the local Arrowhead cloud are not available to other systems outside the cloud. This is the reason why we must add that during this work we have not designed a solution to tackle the relevant data identification topic. Nevertheless, we have generated the basic infrastructure that enables the interoperability and scalability of the solutions. Based on that, the next step and research work will be focused on the area of relevant data which was identified as a challenge to be solved in the problem analysis step.

Acknowledgment

The project has been developed by the Software and Systems Engineering and Data Analytics and Cybersecurity Research Groups of MGEP. IK4-IDEKO, SAVVY and Ulma Embedded Systems and supported by the Department of Education, Universities and Research of the Basque Government under the projects Ikerketa Taldeak (Grupo de Sistemas Embebidos) and TEKINTZE (Elkartek 2018) and the European H2020 research and innovation programme, ECSEL Joint Undertaking, and National Funding Authorities from 19 involved countries under the project Productive 4.0 with grant agreement no. GAP-737459 - 999978918.

References

[1] J. Cecil, S. Albuhamood, P. Ramanathan, and A. Gupta, "(smart CPS) an internet-of-things (iot) based cyber manufacturing framework for the assembly of microdevices," *Int. J. Computer Integrated Manufacturing*, vol. 32, no. 4-5, pp. 430–440, 2019.

[2] A. Rymaszewska, P. Helo, and A. Gunasekaran, "Iot powered servitization of manufacturing–an exploratory case study," *International Journal of Production Economics*, vol. 192, pp. 92–105, 2017.

[3] F. Bonomi, R. A. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing, MCC@SIGCOMM 2012, Helsinki, Finland, August 17, 2012*, 2012, pp. 13–16.

[4] F. Larrinaga, I. Aldalur, M. Illarramendi, M. Iturbe, T. Perez, G. Unamuno, J. Salvidea, and I. Lazkanoiturburu, "Analysis of technological architectures for the new paradigm of the industry 4.0," *DYNA*, vol. 94, no. 3, pp. 267–271, 2019.

[5] J. Sztipanovits, S. Ying, I. Cohen, D. Corman, J. Davis, H. Khurana, P. Mosterman, V. Prasad, and L. Stormo, "Strategic r&d opportunities for 21st century cyber-physical systems," *Steeting Commitee for the Foundation fo Innovation in Cyber-Physical Systems, National Institute of Standards and Technology (NIST)*, 2013.

[6] B. Kamp, A. Ochoa, and J. Diaz, "Smart servitization within the context of industrial user-supplier relationships: contingencies according to a machine tool manufacturer," *International Journal of Interactive Design and Manufacturing*, vol. 11, no. 3, pp. 651–663, 2017.

[7] O. Givehchi, K. Landsdorf, P. Simoens, and A. W. Colombo, "Interoperability for industrial cyber-physical systems: An approach for legacy systems," *IEEE Trans. Industrial Informatics*, vol. 13, no. 6, pp. 3370–3378, 2017.

[8] N. Pedersen, T. Bojsen, and J. Madsen, "Co-simulation of cyber physical systems with hmi for human in the loop investigations," in *Proceedings of the Symposium on Theory of Modeling & Simulation*. Society for Computer Simulation International, 2017, p. 1.

[9] B. Liu, Y. Zhang, G. Zhang, and P. Zheng, "Edge-cloud orchestration driven industrial smart product-service systems solution design based on cps and iiot," *Advanced Engineering Informatics*, vol. 42, p. 100984, 2019.

[10] S. Kim and S. Park, "Cps (cyber physical system) based manufacturing system optimization," *Procedia computer science*, vol. 122, pp. 518–524, 2017.

[11] L. D. Xu and L. Duan, "Big data for cyber physical systems in industry 4.0: a survey," *Enterprise IS*, vol. 13, no. 2, pp. 148–169, 2019.

[12] A. Bagozi, D. Bianchini, V. D. Antonellis, A. Marini, and D. Ragazzi, "Summarisation and relevance evaluation techniques for big data exploration: The smart factory case study," in *Advanced Information Systems Engineering - 29th International Conference, CAiSE 2017, Essen, Germany, June 12-16, 2017, Proceedings*, 2017, pp. 264–279.

[13] A. A. Ahmad and P. Andras, "Scalability analysis comparisons of cloud-based software services," *J. Cloud Computing*, vol. 8, p. 10, 2019.

[14] A. Gilchrist, *Industry 4.0: the industrial internet of things*. Apress, 2016.

[15] L. D. Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Trans. Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.

[16] S. Schneider and R. Joshi, "A practical guide to using the industrial internet connectivity framework," 2017.

[17] E. Borgia, "The internet of things vision: Key features, applications and open issues," *Computer Communications*, vol. 54, pp. 1–31, 2014.

[18] B. D. Martino, M. Rak, M. Ficco, A. Esposito, S. A. Maisto, and S. Nacchia, "Internet of things reference architectures, security and interoperability: A survey," *Internet of Things*, vol. 1-2, pp. 99–112, 2018.

[19] Q. Tan, N. El-Bendary, M. A. Bayoumi, X. Zhang, J. Sedano, and J. R. Villar, "Emerging technologies: Iot, big data, and CPS with sensory systems," *J. Sensors*, vol. 2018, pp. 3 407 542:1–3 407 542:3, 2018.

[20] C. Sanín, H. Zhang, S. I. Shafiq, M. M. Waris, C. S. de Oliveira, and E. Szczerbicki, "Experience based knowledge representation for internet of things and cyber physical systems with case studies," *Future Generation Comp. Syst.*, vol. 92, pp. 604–616, 2019.

[21] T. Krismayer, R. Rabiser, and P. Grünbacher, "A constraint mining approach to support monitoring cyber-physical systems," in *Advanced Information Systems Engineering - 31st International Conference, CAiSE 2019, Rome, Italy, June 3-7, 2019, Proceedings*, 2019, pp. 659–674.

[22] K. Huang, C. Zhou, Y. Qin, and W. Tu, "A game-theoretic approach to cross-layer security decision-making in industrial cyber-physical systems," *IEEE Trans. Industrial Electronics*, vol. 67, no. 3, pp. 2371–2379, 2020.

[23] W. Ding and H. Hu, "On the safety of iot device physical interaction control," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS 2018, Toronto, ON, Canada, October 15-19, 2018*, 2018, pp. 832–846.

[24] A. Forward and T. Lethbridge, "The relevance of software documentation, tools and technologies: a survey," in *Proceedings of the 2002 ACM Symposium on Document Engineering, McLean, Virginia, USA, November 8-9, 2002*, 2002, pp. 26–33.

[25] Y. Pikus, N. Weißenberg, B. Holtkamp, and B. Otto, "Semi-automatic ontology-driven development documentation: generating documents from RDF data and DITA templates," in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, SAC 2019, Limassol, Cyprus, April 8-12, 2019*, 2019, pp. 2293–2302.

[26] P. R. Oliveira, M. Meireles, C. Maia, L. M. Pinho, G. Gouveia, and J. Esteves, "Emulation-in-the-loop for simulation and testing of real-time critical CPS," in *IEEE Industrial Cyber-Physical Systems, ICPS 2018, Saint Petersburg, Russia, May 15-18, 2018*, 2018, pp. 258–263.

[27] D. Marijan, A. Gotlieb, and M. Liaaen, "A learning algorithm for optimizing continuous integration development and testing practice," *Softw., Pract. Exper.*, vol. 49, no. 2, pp. 192–213, 2019.

[28] Y. D. Barve, H. Neema, A. S. Gokhale, and J. Sztipanovits, "Model-driven automated deployment of large-scale CPS co-simulations in the cloud," in *Proceedings of MODELS 2017 Satellite Event*, 2017, pp. 463–464.

[29] M. Alam, J. Rufino, J. Ferreira, S. H. Ahmed, N. Shah, and Y. Chen, "Orchestration of microservices for iot using docker and edge computing," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 118–123, 2018.

[30] P. González-Nalda, I. Etxeberria-Agiriano, I. Calvo, and M. C. Otero, "A modular cps architecture design based on ros and docker," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 11, no. 4, pp. 949–955, 2017.

[31] S. P. Kane and K. Matthias, *Docker: Up & Running: Shipping Reliable Containers in Production*. O'Reilly Media, 2018.

[32] M. Jackson, S. Crouch, and R. Baxter, "Software evaluation: criteria-based assessment," *Software Sustainability Institute*, 2011.