An OSGi-based production process monitoring system for SMEs

(Article begins on next page)

An OSGi-based production process monitoring system for SMEs

Andrea Bonci
dept. of Information Engineering (DII)
Politechnic University of Marche
Ancona, Italy
a.bonci@univpm.it

Alessandro Di Biase
dept. of Information Engineering (DII)
Politechnic University of Marche
Ancona, Italy
s1104529@studenti.univpm.it

Maria Cristina Giannini
dept. of Information Engineering (DII)
Politechnic University of Marche
Ancona, Italy
s1096957@studenti.univpm.it

Marina Indri
Dept. of Electronics and Telecommunication
Politecnico di Torino
Torino, Italy
marina.indri@polito.it

Andrea Monteriù
dept. of Information Engineering (DII)
Politechnic University of Marche
Ancona, Italy
a.monteriu@univpm.it

Mariorosario Prist
dept. of Information Engineering (DII)
Politechnic University of Marche
Ancona, Italy
m.prist@univpm.it

***Abstract*—The present paper proposes an architecture for a product process monitoring system suitable for SMEs (Small-Medium Enterprises). The monitoring system is the main means by which decision-making systems based on intelligent automation technologies are aware of the state of the system on which they will take decisions. Methods and tools from best-practice and best-effort approaches are proposed in the context of SMEs, where the requirements of low cost, low initial level of digitisation and high production flexibility often coexist and contribute to the complexity of management and control problems in these companies. The paper focuses on the design of the monitoring system using an OSGi framework to meet industry standards and Industry 4.0 requirements, taking into account the peculiarities of SMEs as design constraints. The proposed architecture was first tested using a simulation tool and then implemented on a full-scale production line used for data collection.**

***Index Terms*—Monitoring System, OSGi, Cyber Physical systems, IIoT, Industry 4.0, Mobile Robotics.**

I. Introduction

To ensure a satisfactory service level for SMEs' customers, it is essential to carry out proper monitoring of production processes. To monitor production areas and to assess whether the process progress is in line with the production plan, companies implement various systems. Enterprises using information technology (IT) can implement real-time monitoring systems to adjust production plans according to the customer needs, while improving manufacturing resources and reducing costs [1]. Through real-time tracking of the in-process product, the manufacturing process of the shop floor can be improved [2].

Despite the rapid spread of digital technologies and tools, many SMEs monitoring systems are still based on paper documentation, or simply on excel sheets, while the reporting of the activity carried out is done through the use of customised, often non-digital solutions. Indeed, in many SMEs, the production areas are mainly based on manual and/or semi-manual operations and the production machines are of old manufacture, and thus not designed to be able to exchange data with other systems unless they are excessively costly or have to be replaced. On the other hand, large enterprises or more competitive companies use advanced and high-cost IT solutions with dedicated software, such as Manufacturing Execution System (MES), Enterprise Resource Planning (ERP), etc., and are starting to use Industrial Internet of Thing (IIoT) solutions that allow industrial sensors to be connected to new IT architectures. It is worth mentioning that companies often implement hybrid systems or do not have a standardised system for process monitoring. The system that an SME implements may depend on various factors, such as the size of the company, technical and financial possibilities, existing knowledge in the company and so on. Therefore, SMEs need sufficiently flexible and cost-efficient solutions for their size and reality. Some of the most common monitoring systems in SMEs are the following ones;

a) Paper-based monitoring systems - even today, many companies, both small and medium-sized, use monitoring systems based on paper-based reports, such as time-diary measures or the self-reported working time estimate. Some reasons might be related to economic aspects or to the lack of awareness of the benefits of additional technological support. The main drawbacks of these systems is that self-reported work, time diaries or other paper-based techniques have to be analysed by a person with the necessary knowledge and experience to identify and act quickly to solve problems.

b) Computer-based monitoring systems - the implementation of such systems to automatically control production processes without operator intervention are increasingly used in modern companies. Many applications can be found in literature on how to implement an integrated production system

This research is supported by: the ENCORE Horizon 2020 project grant agreement no. 820434; the REACT project, "REACTive Product Design and Manufacturing", funded by italian MISE; "HD3Flab-Human Digital Flexible Factory of the Future Laboratory", ERDF, POR MARCHE region FESR 2014/2020.

using, for example, RFID-based monitoring systems, Internet of Thing (IoT) [3], or the most advanced cyber-physical systems [4]. Further applications can be found in various industries, both manufacturing [5], [6] and non-manufacturing, such as the oil industry [7], process industries of cement [10], food industry [8], waste industry [9], and others.

c) IIoT-based monitoring systems - As IoT is an extension of the Internet and other network connections to various devices, sensors or things, IIoT is the technology that deals with complex physical systems connected via industrial sensors [11]. IIoT-based monitoring systems extend computer-based ones to include distributed smart objects [12]. However, the IIoT takes this to a new level by including machine learning, predictive maintenance [13], and big data technology along with industrial supervision; it also improves production efficiency, supply chain management, quality control, sustainability and green practices [14].

The monitoring systems proposed in the literature can refer to a complete integration of information and material flow or they can refer to selected sub-phases of production process, e.g., in [15] the authors focus on the monitoring of a single phase of the manufacturing process, i.e., the cutting one. In [16], the authors focus on machines condition monitoring, while in [17] the author presents an integrated monitoring system for discrete manufacturing processes which integrates information flow of the whole production process. Examples of production monitoring systems dedicated to SMEs can also be found in literature, for example, in [18], but the system excludes any human intervention in the data collection process, and this is often not feasible in most SMEs. Another example can be found in the solution developed by Bosch for large enterprises; the Bosch IoT OSGi-Based Edge Services (formerly Bosch IoT Gateway Software), which allows a range of IoT edge devices to be connected to process and manage device data locally to make fast, intelligent decisions at the edge, or move normalised data for further analysis to the cloud [19]. Although it is possible, as pointed out above, to find similar intelligent application frameworks with many advanced functions, they are too expensive, inflexible and complex for SMEs. Therefore, the following objectives have been pursued in this work: a simple production process monitoring system based on the OSGi framework is proposed and developed with the aim of being implementable with as little effort as possible, to be integrated with heterogeneous automation systems, even those that are not state-of-the-art, and at the same time will have to provide the information required to manage the production process of complex products. In fact, the monitoring system based on OSGi framework allows to add different type of sensors, algorithms, views, etc. in a plug and play way with low effort saving time and costs. The remainder of the paper is organized as follows. Section II presents the architecture which poses the basis for the proposed solution, including a brief introduction about OSGi framework and the developed software modules to manage heterogeneous types of sensors, real and simulated. The pilot case and the preliminary result are described in Section III.

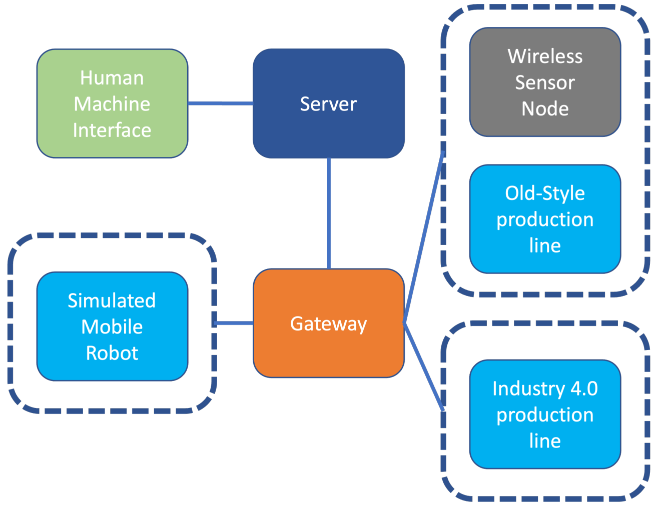


Fig. 1. General architecture blocks diagram. Main actors.

Last part summarizes the work, presents its limitations and describes future work.

II. Architecture of OSGi-based monitoring system

As mentioned above, a monitoring system for SMEs have to be easy to install, modular, maintainable, and low cost, with the possibility to integrate different devices in plug and play manner without having to write numerous lines of code. To this end, the focus was placed on a modular, extensible and inter operable software for managing heterogeneous devices, based on OSGi framework. Furthermore, it is important to emphasise that the proposed solution does not want to establish a standard for the SMEs, such as RAMI [20] or other industry-standard solutions, but to highlight how the use of the OSGi framework in the context of Industry 4.0 can meet the above-mentioned requirements for SMEs. As a proof-of-concept solution, a PC-based gateway with OSGi runtime was used to load all developed modules. Furthermore, in order to describe the whole software architecture, a preliminary description of its basic modules is provided.

Although this approach can be used to implement a monitoring system for any type of heterogeneous automation system, this work focused on system integration problems that are encountered in SMEs, namely an industrial production line conforming to Industry 4.0 standards, an old-fashioned production line using an added wireless sensor node for data communication, and an autonomous mobile robot to share products and materials between production lines. In addition, at this early stage, a basic web dashboard was designed for data visualisation. The architecture of the proposed monitoring system and the main actors are summarised in Figure 1. Conceptually, the scheme of a simplified monitoring system for a typical SME consists of two main parts, namely a group of devices to be integrated and a gateway. The gateway module is the heart of the system, and it will be described in detail in this section, after the introduction of the main features of the OSGi framework, whereas the modules of hardware architecture will be illustrated in Section III, with reference to the considered pilot case.

A. OSGi Framework

In 1999, IBM, Oracle, Philips, Sun, and other companies founded the non-profit Open Service Gateway initiative (OSGi) with the aim to harmonise and to standardise the development processes of modular applications in automotive, home automation and industrial automation field. OSGi defines a dynamic module system for the Java language that allows modules to be defined and how they interact with each other at runtime using a centralised service-oriented architecture with loosely coupled services [22]. In summary, an OSGi architecture is composed by OSGi Framework, a set of bundles and a set of relationships. More details for OSGi programming and developing applications can be found in [23]. OSGi framework has several advantages, platform Independence, application Independence, dynamic Updates, service Collaboration Support, etc. All these features are useful and desirable for monitoring systems in SMEs. On the other hand, some of its drawbacks are mainly related to programming aspects [24], such as limited built-in support for distributed services, limited capabilities of service registry and lack of shared semantics. Despite these technical disadvantages, the benefits for the needs of SMEs are attractive.

B. Gateway

The Gateway is an embedded device, or a PC-based device, used to interconnect different types of IIoT devices, like PLC, sensors, web-services, etc. with a supervisory system. The Gateway communicates with the external devices through drivers, such as PROFINET, Modbus, Highway HART, EtherCAT, etc., so the Gateway can bee seen as a bridge for the bidirectional data communication between the field devices and the supervisor. In addition to this, considering the vast amount of data to be managed, it should implement different levels of efficiency and security policies. In this context, the modular programming approach proposed to design a Gateway run-time, allows the rapid integration of different types of sensors, algorithms, views, etc. in a plug and play way through the addition of new software modules with low effort saving time and costs. The application developed using the OSGi framework has an extensible core to control data from multiple bundles that cooperate through a service communication architecture. The proposed approach consists of a supervision bundle and a set of bundles that are linked to sensors, devices, algorithms, etc. or used for database management, external communication, as well as notification mechanisms. The supervisor bundle constitutes the active extensible core used to load all bundles in the software platform. The supervisor bundle can be managed either directly in the Gateway, with local software programming, or remotely, via web platform which provides a set of tools for bundles administration. The Bundles architecture includes different type of sensors and devices in order to extend the functionality of the proposed monitoring system. Note that, in the bundles block diagram, there are not explicit and direct references to bundles linked to sensors, but only bundles that implement a standard or private communication protocol interface, making the devices integration further transparent to the system. A sensor can be passive and communicate in polling mode with the supervisor, active by sending its data regularly to the supervisor, or in service mode with the publisher/subscriber paradigm. Another important bundle is the local database manager, used to save the data produced by bundles interfaced with the sensors. However, the database manager is primarily intended to be used by other bundles, which access the database manager using a manager client interface. Together, database manager and manager clients offer a consistent, standard interface to data that also handles inter-process communication and secure data access. A database manager coordinates access to the data storage layer in the application for a number of different interfaces. The local database includes static tables to store the collection of every device used in the system and dynamic tables to contain processed data and, eventually, statistics. From the data analysis point of view, it is possible to load bundles dedicated to specific algorithms. As shown in Figure 2, an asynchronous scheduler bundle manages the algorithm to be used and when to run it, according to the information stored in the database. The scheduler bundle generates events. Events are linked to tasks that execute on a specified time and can be scheduled. The number of tasks have to be started is directly proportional to the number of different automation systems installed and to the number of algorithms to be monitored. Generally, a task is directly started only if there is an algorithm to be calculated. In order to reduce the complexity and increase the efficiency only a reduced number of thread in runtime has been managed. In fact, a task can have multiple processes occurring at the same time while threads can only have one task running at a time. The proposed bundles schema for the asynchronous scheduler bundle is composed by three bundle: the Scheduler, the Dispatcher and, finally, the Threshold. While the scheduler module publishes a new message when each event is verified, the Dispatcher module decides if the

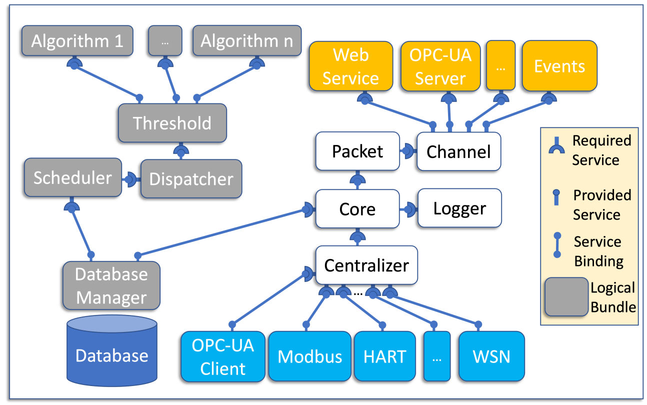


Fig. 2. Bundles block diagram and services relationship.

current event can be related to a specific algorithm or sensor, and it notifies to all subscribed services to run a specific data processing technique; the Threshold bundle verifies if the published message payload contains a value above the threshold.

III. Pilot Case and Preliminary Results

In this section, the focus is placed on the pilot case of a real complex industrial automation system composed by different actors. The evaluated tests have been placed to verify the feasibility, the modularity and the extensibility of the solution based on the OSGi Framework, the correct visualization of information and the definition of the future services through data analysis algorithms. Although it can be possible to use only one device at time, it is valuable to verify the behaviour of the whole architecture including different types of real automation systems and wireless sensors. The main goals to reach in the test stage were: Monitoring of a production area and integration of new bundles in OSGi Framework with a dedicated communication protocol in plug and play mode.

A. Pilot Case description

The pilot case has been composed by three main parts, the assembly line (Cyber Physical Factory by Festo), the Wireless Sensor Node by Advantech and an autonomous mobile robot. these parts are briefly described below.

1) Assembly line: The assembly line around which the pilot case has been constructed is an important part of i-Labs industry, a Public-Private-Partnership laboratory in Italy which is devoted to technology transfer and cooperation between industry and academia. The lab has production aid systems, such as collaborative and mobile robots and human-compliant automation systems, as well as IIoT and data analysis systems, capable of advanced management of production processes. The assembly line by Festo simulates the assembly process of the main components of an electronic device (e.g. assembly of front cover, back cover and a printed circuit board). The production line, shown in Figure 3, is composed of four main modules (assembly modules, automatic warehouse for carriers, a manufacturing module, a quality control module and a manual station) connected to each other by a conveyor. Each manufacturing module or workstation is dedicated to a specific product operationthat can be managed by a Manufacturing Executing System 4.0 (here the Festo MES4). As shown above, the assembly line comprises four modules. Each is directly connected to the next and there are no buffers to handle more than eight products at time. Although each module can only handle one product at a time, the length of the transfer line allows more than one product to be handled at each station. The carriers move around the line by conveyors and in relation to the planned activity they can be stopped at each working station. A wide variety of IIoT sensors, PLCs and Digital Twins allow to control and to verify the production progress. Even though it is possible to manage automatically the control of the CP-Factory operations connecting with the MES, in our test case a direct link with the available OPC-UA servers has been implemented (each module of the production line has its own Siemens PLC and OPC-UA server). The integration of the Festo modules through the SIEMENS OPC-UA servers is based on a OSGi Bundle which implements from one side a four OPC-UA client instances (polling mode) and from the other side an interface to publish data in a dedicated channel.

Fig. 3. CP-Factory, a smart and modular assembly line by Festo.

2) Wireless Sensor Node: The Wireless Sensor Node (WSN) used in the proposed scenario is the WISE-4012 by Advantech, which is an IoT wireless I/O module with 2 analog input channels, 2 digital input channels and 2 digital relay output (See Figure 4). The WISE-4212 is physically installed on a CP-Factory line but the input channels have been connected to an extension board for simulating sensors status of an old-style machine in order to check the correct functionality of the data communication level, Industrial Wi-Fi interface and data acquisition. The WISE-4000 series supports two types of communication protocol, the Modbus/TCP or the RESTful web service (standard IoT communication protocol). Data can be polled or even be pushed automatically from the node when the I/O status is changed. Therefore, since the wireless sensor node can provide two communication ways to send and receive data, an OSGi module has been developed to allow the integration of the node in a plug-and-play way. In this case, it was chosen to develop a OSGi Bundle for the RESTful web service protocol instead of Modbus/TCP since it

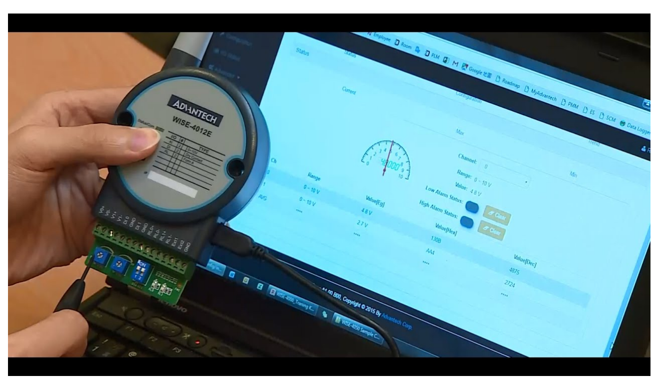


Fig. 4. Wireless Sensor node WISE-4012e and dashboard web for wireless configuration.

does not implement a secure communication layer. The OSGi module has two interfaces, one of which to manage REST communication and the other interface to publish data on the dedicated channel.

3) Autonomous Mobile robot: The presence of an Autonomous Mobile Robot (AMR) is included, considering it as an intelligent agent capable of actively interact with the industrial environment, enhancing the flexibility of the proposed monitoring architecture. It is assumed to be equipped with a standard 2D lidar sensor and to be able to autonomously navigate in the manufacturing environment, using the navigation stack included in the ROS2 software platform (See Figure 5). The AMR can contribute in different ways to the monitoring task, since it can be equipped with specific sensors able to provide the required information to investigate the status of each assembly module or station (e.g., not only standard vision sensors, but also depth or thermal cameras, etc.). This way it can be seen as a mobile meta-sensor. In [25], [26], the Sen3Bot Net architecture was sketched to allow safe navigation of AMRs in an industrial environment shared with human operators. Here, the AMR is employed to enhance the flexibility of the monitoring architecture, reducing the necessity of installing several fixed sensors of different types for each station: only a reduced set of sensors is located in a predefined way in the environment infrastructure, whereas the AMR is moved along the production line, according to a pre-planned repetitive path, in order to periodically check the status of all the stations. In case of a bottleneck detected by one of the fixed sensors, the AMR can be sent on demand to the particular assembly module involved, to acquire further information about the possible reasons of the problem. Such a solution is useful also to include in the monitoring architecture *old* machineries that are not equipped by proper sensors or that cannot be easily connected through the wireless communication network, and facilitates a possible reconfiguration of the manufacturing process in SMEs, where changes are more frequent than in the production lines of large companies. The AMR is included in the wireless sensor network handling the entire monitoring system, according to the adopted ROS2 platform, which uses Data Distribution Services (DDS) as communication standard. From the perspective of communication middleware, an OPC-UA publisher/subscriber-based node has then been implemented within the OSGi Framwork in order to exchange data with other architecture actors. The kind and amount of exchanged data depend on the specific task assigned to the mobile robot.

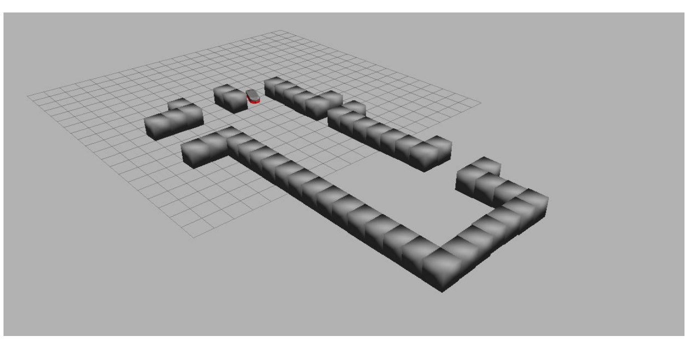


Fig. 5. Simulation of the real laboratory environment via GAZEBO.

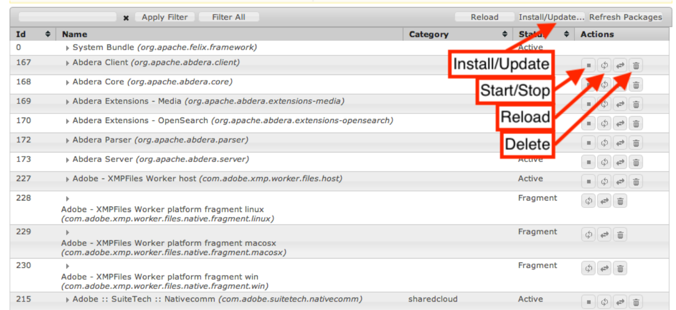
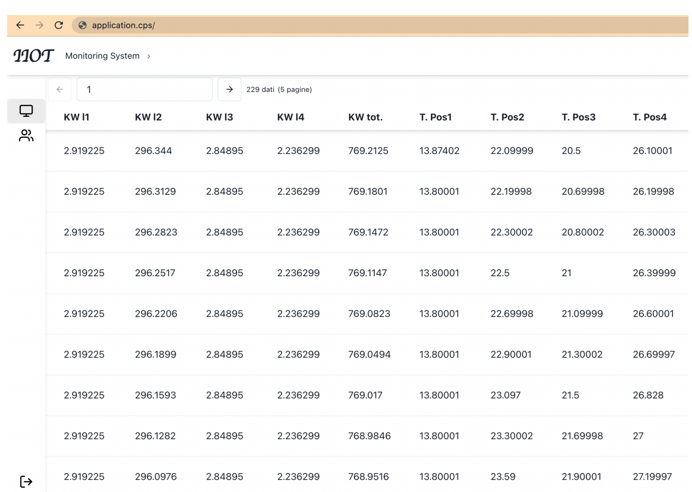


Fig. 6. OSGi console with commands tool.

B. Preliminary results

In this early stage of development of the proposed solution, it is valuable to implement new OSGi bundles, which from one hand exchange data with the coordinator bundle and on the other hand use a standard communication between the external interface (OPC-UA, Web-service, etc.) and the gateway. The goal is to validate the concept of modularity, flexibility and interoperability of the developed software architecture and to define a standard for the monitoring system configuration process in order to deploy the same solution on other SMEs with different assembly lines or manual workstations (different PLCs, WSNs and communication protocols). The adaptability feature, due to the OSGi framework, permits to build one or more bundles associated to a different automation systems to integrate. An overview of OSGi console where new bundles can be uploaded, modified bundles can be reloaded or eventually only started and stopped is highlighted in Figure 6. In our scenario, some bundles periodically retrieve data in a polling mode, while other ones use publisher/subscriber approach;



KW I1	KW I2	KW I3	KW I4	KW tot.	T. Pos1	T. Pos2	T. Pos3	T. Pos4
2.919225	296.344	2.84895	2.236299	769.2125	13.87402	22.09999	20.5	26.10001
2.919225	296.3129	2.84895	2.236299	769.1801	13.80001	22.19998	20.69998	26.19998
2.919225	296.2823	2.84895	2.236299	769.1472	13.80001	22.30002	20.80002	26.30003
2.919225	296.2517	2.84895	2.236299	769.1147	13.80001	22.5	21	26.39999
2.919225	296.2206	2.84895	2.236299	769.0823	13.80001	22.69998	21.09999	26.60001
2.919225	296.1899	2.84895	2.236299	769.0494	13.80001	22.90001	21.30002	26.69997
2.919225	296.1593	2.84895	2.236299	769.017	13.80001	23.097	21.5	26.828
2.919225	296.1282	2.84895	2.236299	768.9846	13.80001	23.30002	21.69998	27
2.919225	296.0976	2.84895	2.236299	768.9516	13.80001	23.59	21.90001	27.19997

Fig. 7. Dashboard data visualizations.

through a bundle dedicated to manage heterogeneous type of data, all data are then saved in a buffer, and finally sent to the supervisor Bundle, as a formatted packet including all the buffer information. The pilot case and the proposed solution were tested for a period of one months in order to verify the stability of the software and functionalities, such as stopping, starting, loading and deleting all OSGi bundles, as well as device set-up, software configuration and web dashboard display. The Figure 7 shows some data acquired from different types of sensors in terms of power consumption (all motors used in the CP-Factory) and temperature modules (WSN). Once the software architecture has been developed and the bundles used to connect the system to the sensors have been tested, each installation differs from the first only in terms of configuration and data display. The developed code can be completely reused and only a few days are needed to complete the procedure. Of course, our focus is on the monitoring system, which means that data analysis is delegated to the upper layers of the automation pyramid.

IV. Conclusion

Industry 4.0 brings the SMEs to a new level by introducing modular, flexible and customized production technologies. This means that manual workstations and assembly lines will be integrated in all decision making processes in order to create an optimized complex system. Following the paradigm of new smart factories, SMEs have to implement a monitoring system in order to monitor production areas and to assess whether the process progress is in line with the production plan. In this paper has been presented a new modular, extensible and interoperable software monitoring system based on OSGi framework with the aim to integrate heterogeneous devices, simulation and IoT devices in a simplified and low cost way. The proposed architecture has been tested on a pilot case composed by CP-Factory by Festo, an autonomous mobile robot based on ROS2 and a wireless sensor node using WISE-4012 by Advantech, for two weeks, in order to verify the software stability and the replicability of the proposed solution in other scenarios in a plug and play way. Further activities, regarding functional integration and sensing, will be carried out to increase performance of the proposed system.

References

[1] R. Y. Zhong, X. Xu, E. Klotz, and S. T. Newman, "Intelligent Manufacturing in the Context of Industry 4.0: A Review," Engineering, vol. 3(5), pp. 616–630, 2017.

[2] S. Jaskó, A. Skrop, T. Holczinger, T. Chován, and J. Abonyi, "Development of manufacturing execution systems in accordance with Industry 4.0 requirements: A review of standard and ontology-based methodologies and tools," Computers in Industry, vol. 123, 2020.

[3] W. Chen, "Intelligent manufacturing production line data monitoring system for industrial internet of things," Computer Communications, vol. 151, pp. 31—41, 2020.

[4] A. Bonci, M. Pirani, and S. Longhi, "A DataBase-Centric Framework for the Modeling, Simulation, and Control of Cyber-Physical Systems in the Factory of the Future" Journal of Intelligent Systems, vol. 27, no. 4, pp. 659-679, 2018.

[5] J. Xu, J. Yao, L. Wang, Z. Ming, K. Wu and L. Chen, "Narrowband Internet of Things: Evolutions, Technologies, and Open Issues," IEEE Internet of Things Journal, vol. 5, pp. 1449—1462, 2018.

[6] D. Stadnicka, A. Bonci, M. Pirani and S. Longhi, "Information management and decision making supported by an intelligence system in kitchen fronts control process", Int. Conf. on Intelligent Systems in Production Engineering and Maintenance ISPEM 2017, Publ. Springer, Advances in Intelligent Systems and Computing, vol. 637, pp. 249-259, 2018.

[7] W. Fang, H. Jiang, J. Li, W. Miao, K. Ma, and W. Xiao, "An intelligent production fluctuation monitoring system for giant oilfield development," Journal of Embedded Systems, vol. 9, pp. 36—44, 2017.

[8] F. Ying and L. Fengquan, "Application of Internet of Things to the Monitoring System for Food Quality Safety," 2013 Fourth Int. Conf. on Digital Manufacturing and Automation, 2013, pp. 296-298, doi: 10.1109/ICDMA.2013.71.

[9] S. Longhi, D. Marzioni, E. Alidori, G. DiBuò, M. Prist, M. Grisostomi, and M. Pirro, "Solid Waste Management Architecture Using Wireless Sensor Network Technology," 5th International Conference on New Technologies, Mobility and Security, 2012.

[10] S.M. Zanoli, L. Barboni, F. Cocchioni, and C. Pepe. "Advanced process control aimed at energy efficiency improvement in process industries", Proceedings of the IEEE Int. Conf. on Industrial Technology, February 2018, pp. 57-62.

[11] C. Gong, "Human-Machine Interface: Design Principles of Visual Information in Human-Machine Interface Design," International Conference on Intelligent Human -Machine Systems and Cybernetics, vol. 2, 2009.

[12] J. S. Rha, and HH. Lee, "Research trends in digital transformation in the service sector: a review based on network text analysis," Service Business Springer, vol. 16, pp. 77–98, 2022.

[13] A. Bonci, S. Longhi, G. Nabissi, and F. Verdini. "Predictive Maintenance System using motor current signal analysis for Industrial Robot" IEEE Int. Conf. on Emerging Technologies and Factory Automation, ETFA 2019, pp. 1453-1456.

[14] A. Karmakar, N. Dey, T. Baral, M. Chowdhury and M. Rehan, "Industrial Internet of Things: A Review," International Conference on Opto-Electronics and Applied Optics, vol. 3, pp. 1–6, 2019.

[15] E. Koziowski, D. Mazurkiewicz, T. Zabinski, S. Prucnal, and J. Sep, "Assessment model of cutting tool condition for real-time supervision system," Eksploatacja i Niezawodnosc – Maintenance and Reliability, vol. 21(4), pp. 679–685, 2019.

[16] S. Guo, W. G. Guo, A. Abolhassani, R. Kalamdani, S. Puchala, A. Januszczak, and C. Jalluri, "Manufacturing Process Monitoring With Nonparametric Change-Point Detection in Automotive Industry," Journal of Manufacturing Science and Engineering, vol. 141(7), pp. 25–32, 2019.

[17] P. Oborski, "Integrated monitoring system of production processes", Management and Production Engineering Review, vol. 7(4), pp. 86–96, 2016.

[18] A. Snatkin, K. Karjust, J. Majak, T. Aruväli, and T. Eiskop, "Real time production monitoring system in SME", Estonian Journal of Engineering, vol. 19(1), pp. 62–75, 2013.

[19] IOT Suite - OSGi Framework. [online], Available: https://docs.bosch-iot-suite.com/edge/, last accessed May. 2022

[20] Platform Industrie4.0, Reference Architectural Model Industrie 4.0 (RAMI 4.0). [online], Available: https://ec.europa.eu/futurium/en/system/files/ged/a2-schweichhart-reference_architectural_model_industrie_4.0_rami_4.0.pdf, last accessed May. 2022

[21] R. Langmann, and L. F. Rojas-Peña, "A PLC as an Industry 4.0 component", Proc. of 13th International Conference on Remote Engineering and Virtual Instrumentation, pp. 100–155, 2016.

[22] C. Teng-Wen, "Android/OSGi-based vehicular network management system", Proc. 12th International Conference on Advanced Communication Technology, pp. 1644–1649, 2010.

[23] The OSGi Alliance, OSGi Service Platform Service Compendium. [online], Available: https://www.osgi.org, last accessed May. 2022

[24] J. Rykowski, and D. Wilusz "Comparison of architectures for service management in IoT and sensor networks by means of OSGi and REST services", Federated Conference on Computer Science and Information System, pp. 1207–1214, 2014.

[25] M. Indri, L. Lachello, I. Lazzero, F. Sibona, and S. Trapani, "Smart Sensors Applications for a New Paradigm of a Production Line," Sensors, vol. 19, n.3, 650, 2019.

[26] M. Indri, F. Sibona, and P.D. Cen Cheng, "Sen3Bot Net: a meta-sensors network to enable smart factories implementation", 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2020), Vienna, Austria, pp. 719-726, 2020.