

A Smart web-based Maintenance System for a smart manufacturing environment

Ana M. Alexandru, Alice De Mauro, Maurizio Fiasché, *Senior, IEEE*, Francesco G. Sisca, Marco Taisch, Luca Fasanotti, and Piergiorgio Grasseni

Abstract— Maintenance is a practice in manufacturing that had never been available to remote control and management until the introduction of web-connected portable smart devices. In the last years several studies and applied research have been conducted for achieving this objective in an efficient way and with the aim to enhance the business activity related to. Remote access and role-specific data distribution can become the next level upgrade of maintenance, diagnostic and flow control management using smart sensors, actuators, and smart consumer devices (smartphone, tablet, etc.).

In this project, a real case is presented, an Italian company, the end user of the project, tried to achieve this goal creating with the all consortium, a new web-services based server application in order to have remote access to the data stream, which permits to have the machine status available on the web, very strict time responses, a better user profiling and innovative control system based on smart devices monitoring real time machine data and sending notification sounds when needed. The result is a platform connecting, using the Internet of Things (IoT) paradigm, industrial machineries with a smart device android app and with a web application running on a normal browser.

Index Terms— IoT; Internet of Things, smart maintenance; smart monitoring; smart devices; web service

I. INTRODUCTION

Maintenance is a goal that had been reached in various ways in the past, but had never been available to remote control and management until the introduction of web-connected portable smart devices. Next goal should be bring such information from industry to the right user in a more flexible and portable way, in order to permit control from every place and easy worker mobility.

Data from production flow should be collected and rightly distributed to be accessible from various kind of users from different places at different moments: a flexible remote access

Submitted on the 25th of May 2015. This work was supported by the TOUCHPLANT Project of Regione Lombardia, Grant number 42703196.

A. M. Alexandru, A. De Mauro, M. Fiasché, F. G. Sisca, M. Taisch are with Politecnico di Milano, Department of Management, Economics and Industrial Engineering, Milano, Italy (e-mail: anamaria.alexandru@polimi.it, alice.demauro@polimi.it, maurizio.fiasche@polimi.it, francescogiovanni.sisca@polimi.it, marco.taisch@polimi.it).

L. Fasanotti is with Intellimech consortium, KM Rosso, Bergamo, Italy and with Università degli studi di Bergamo, Bergamo, Italy (e-mail: luca.fasanotti@intellimech.it).

Piergiorgio Grasseni is with SMILAB, SMITEC S.p.A., San Pellegrino Terme, Bergamo, Italy (e-mail: piergiorgio.grasseni@smigroup.net).

to data should be necessarily standardized.

The maintenance and diagnostic information must also be made available as role-specific information and must be distributed accordingly to each user, which is supplied with specially tailored information that he needs for his tasks.

Remote access and role-specific data distribution can become the next level upgrade of maintenance, diagnostic and flow control management.

The research project has been realized into a project called Touchplant, a regional Italian project funded by Lombardia Region, which aim is to utilize smart sensors, actuators, and smart consumer devices (smartphone, tablet, etc.) in systems to better monitor production processes and machinery condition to streamline production and make it more flexible and efficient.

In short, this project's objective is the development of an integrated economic system for remote configuration, monitoring, and diagnosis of machines and production chain plants in a real case.. Trials were made on a company producing customized assembly production lines, so the topic's complexity (the construction of manufacturing machines) leads to innovative solution strategies In this paper the architecture implementation of the smart maintenance system designed is presented with results of performance and robustness. The rest of the paper is organized in the following way: in section II the problem surround is explained with trends and the as-is in the case of study, in section III is shown the end-to-end architecture, sections IV, V, VI and VII explain in-debt overview about the architecture, in section VIII is cleared the role-specific clustering of data and in section IX and X are shown results and conclusions.

II. SMART SYSTEMS IN MAINTENANCE

The last years have seen quite a few technological innovations in the field of industrial ICTs, belonging to the so called Industry 4.0: Internet of Things is emerging as a new generation of the fully interconnected world (from connection of computer to the connection of devices); Service Oriented Architecture approaches will disclose new opportunities to allow quick implementation of new systems; personalized devices such as smart phones or tablet can improve workers conditions by providing extensive knowledge in the factory [4][5][6].

A. Technological megatrends

Technological megatrends in collaboration, mobility, connectivity and intelligence would empower future manufacturing enterprises to build innovative products and services. In particular:

- **Dynamic collaboration:** efficient and secure collaboration between many different stakeholders will become crucial for day-to-day operations of European manufacturers. Large companies as well as Small Medium Enterprises (SMEs) stand to gain from collaborative manufacturing, service management and customer engagement via social media and other tools.
- **Enterprise mobility:** the exponential proliferation of mobile devices presents an attractive proposition to ‘on-the-go’ and ‘always-on’ users. While mobile technologies have permeated the consumer market, enterprise applications are still relatively limited. To leverage the potential of next-generation smart phones and handhelds, manufacturing enterprises are looking beyond conventional desktop solutions and focus on new opportunities and businesses in the mobile world.
- **Real-world connectivity:** sensors, automation controllers and embedded systems are already commonplace in industrial applications. However, so far few companies have been deploying more than their own Intranet of Things focused on local, isolated and closed-loop scenarios. The trend is to seamlessly and bi-directionally interact with real-world objects and systems on a global scale, across a variety of application domains and stakeholders in a secure way, thus realizing the Internet of Things [1].
- **Manufacturing intelligence:** collaboration and connectivity will give rise to copious amounts of context and data that will have to be analyzed on-the-fly and rendered on mobile devices of decision makers at both management and plant levels. Manufacturing enterprises will have a competitive advantage over their peers if they are able to perform real-time analysis over a large volume of data from processes, products and business systems.

Nowadays, companies have to consider efficient use of energy and resources in manufacturing besides traditional performances to become and remain competitive in their respective industry. In fact, the advantages of adopting energy efficiency solutions can be summarized in: direct costs reduction, regulations respect, processes at low energy consumption, positive product image [2]. Techniques and tools to support and implement energy efficiency in production management are still required and different topics could be analyzed including monitoring, measuring, collecting, analyzing, controlling, and scheduling of material and energy flows of the entire production system [3].

Recently, diagnostics has evolved from fixed time intervals toward condition based policies that reduce the non-useful

maintenance activities but do not prevent situations in which the system operates outside the optimal conditions or it may fail [7]. Then next step is proactive diagnostics that will provide a continuous optimal adjustment of the working conditions optimizing both system durability/efficiency and product/service quality [8].

The so-called “smart maintenance” will provide innovative contribution to improve maintenance activity, in comparison with the current state of the art as explained by the following points:

- presently solutions for predictive maintenance are specialized, while customization is hard to achieve and costly. A “smart maintenance” platform enables making predictive maintenance solutions easy to customize, offering standard solutions for predictive maintenance and a basis to add advanced solutions according to user needs;
- nowadays, software for maintenance analysis are proprietary systems and integration with different maintenance software suites is difficult. Smart maintenance platform will provide, besides standard services/solutions for predictive maintenance and other analysis, basic services allowing the interfacing with other enterprise systems.
- today, in the market, condition monitoring solutions have basic detection features, usually pre-programmed and tight to specific monitoring targets. This makes it hard to directly link detection with diagnosis, as well as to apply existing solutions to multi-point measurement condition monitoring tasks. Smart maintenance will deliver a platform supporting an intelligent grid for distributed novelty detection, a versatile point-of-access for detection reporting and an internal system actor for triggering diagnostic workflows at higher system levels. Such positioned knowledge will effectively build what is called “machinery self-awareness”.

B. Partner as-is and trials

The partner is studying, designing and building machines and modules for the automation of assembly processes. An obsolete standalone software installed on the industrial PC situated on every single machine manages maintenance and diagnostic: based on Visual Basic 6 programming language and an Access database, the software has been developed in almost ten years, becoming a full-optioned software.

Deep personalization and long-term development led to less flexibility, less scalability and harder code management: there are no guides or help to technical support service, there is no remote access to machine, no indication of alarm type, correlation between signals and machine behavior is done asynchronously through observation, each variable is identified by a symbol.

The requirements detected after a functional analysis to create a new application were:

- a better user profiling, permitting to cluster data into different streams sent to each different user according to a given role-specific data use definition (administrator, production manager, maintenance engineer and operative personnel);
- very strict time responses, due to data sampling frequency from the Programmable Logic Controller (from now PLC): about 100ms per single value, hardly reached by the database saving time and the Wi-Fi connection, according to most of the standards;
- remote access to the data stream, which permits to have the machine status available on the web, in an IoT thinking model;
- innovative control system based on smart devices monitoring real time machine data and sending notification sounds when needed.

III. END-TO-END ARCHITECTURE

The architecture model, Fig. 1, as structured in a previous requirements definition, has been defined end to end by a “from machine to remote user” approach.

The data flow model starts from the machine, which can be considered a group of plenty analogic data sources, called sensors. This large group of sensors is structured by and connected to a PLC, which aim is to organize data in a structured and digital way and to send by LAN connection all data to an industrial PC, in which is running a gateway. The gateway manage the data-exchange machine protocol, collecting all sensors’ data and setting the server configuration and communication, sending the data stream to server after previous configuring process, aimed to set properties.

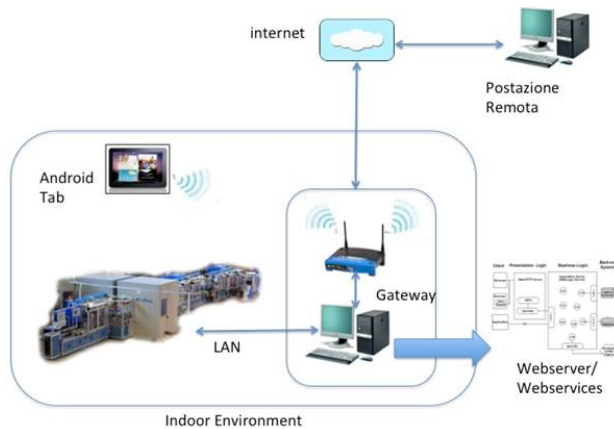


Fig 1. The environment model

After receiving the data, the previously configured server is able to organize them, sending the data stream directly to the web service real-time interface when data are marked as real-time data and to a database when data are marked and collected to be saved. As a peculiarity, data are marked to be “real-time”, so presented to the web service interface and benefited from a remote device using Wi-Fi connection, or are collected and aggregated by a minimum time slot and saved as a database record, forming the data history. At this point, two

different data flows are presented as a web services to two user-based different web application: the first one, Wi-Fi connected and installed on an Android 10.1 tablet, receives the real time data in order to be useful for instant diagnostic, carried on by the maintainer. The second one is a remote access web application, accessible from the web, managing and showing the saved data history from the database, in order to be useful to the production manager in a maintenance and optimization point of view.

IV. FROM MACHINE TO GATEWAY

In this section, the low-level part of the developed architecture is analyzed; in particular, the scope of this segment is to provide the raw data necessary to the front end of the application.

This involves the machine, the gateway and the communication between this two elements performed using MODBUS TCP.

A. Machine

The test machine is a full automatic assembly machine controlled by a single PLC; each assembly station is composed by different equipment linked to the PLC in two different ways: directly linked to the I/O modules of the PLC or linked on the PROFIBUS fieldbus of the machine.

All data are stored in the PLC memory in different data block depending on the type.

TABLE I. PLC MEMORY ALLOCATION

DB200	Real time data, parameter, Warning, Alarm
DB11	Recipe Time
DB12	Recipe Counter
DB105	Recipe String
DB104	Recipe threshold
DB102	Transducers Value And Parameter

The machine-to-gateway communication is based on a MODBUS TCP standard with the support of a dedicated DB on the PLC memory DB411 called Modbus DB.

The PLC has been configured to act as MODBUS server and wait for the data requests from the gateway.

During the working cycle, all data are copied in a specific positions of the Modbus DB in order to merge all the information and assign each parameter to a specific Modbus address.

Furthermore, data that can be written by the application are mirrored in another DB that acts as a shadow copy in order to identify the remote writes.

In the following figure a schema of the working cycle is presented.

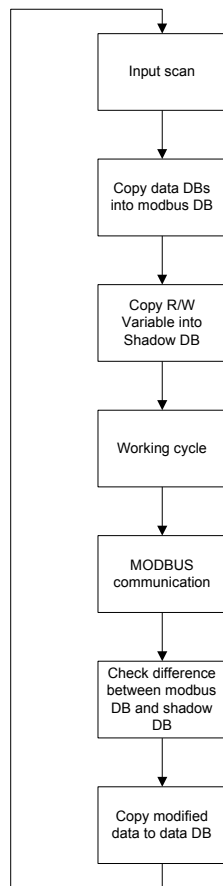


Fig 2. PLC Working Cycle with MODBUS communication

B. Data Transmission

At physical level (PHY), the communication is supplied using a dedicated communication processor of the PLC and a dedicated Ethernet Interface in the gateway linked through a standard CAT 5.e Ethernet cable.

C. Gateway

The gateway has been developed as a custom JAVA application (called UGS) executed into the industrial PC used for trials.

UGS application handles the communication with the machine, acquiring the data and sending the changed ones to the server by using a list of custom web services.

In order to accomplish all specific tasks needed, the application implements a standard MODBUS TCP client that operates on a dedicated network interface.

For machine data acquisition, UGS adopts a polling strategy, querying the PLC every 500ms, acquiring all the data using a read multiple register MODBUS command.

Acquired data are compared with a local copy in order to identify the changed one; if variations occur, they are notified to the server using web services; the behavior previously explained has been specifically chosen to decrease the

computational effort of the PLC, that could cause a relevant decrease of the machine performance, and to reduce the data exchanged with the server.

This point is very important mainly because the server-gateway communication has not been made by using a dedicated network but sharing the network infrastructure of the plant that could be overloaded due to data from other system.

V. FROM MACHINE TO PLC

PLC (a SIEMENS S7 PLC) controls the machine processes and its whole working. It executes the program logic of the machine according the input read from field (sensors, switches, buttons, etc.), and consequently activates output (motors, actuators, lamp, etc.). PLC stores in its memory data, variables, counters, timers, parameters regarding the machine process and this information is accessible from the gateway.

VI. FROM PLC TO GATEWAY

PLC and Gateway exchange data using MODBUS/TCP, an open and standard industrial Ethernet fieldbus: the integration with industrial automation on machine side is facilitated. PLC controls machine and supplies the process data, variables and I/O state in MODBUS/TCP register data unit, read and written by the gateway according to its configuration data list.

The configuration data list allows the declaration of which kind of data will be exchanged on end application side from the gateway to the PLC. Further, it allows declaring which data the Gateway will exchange with the server. The gateway is a JAVA application, executable in Windows, Linux or other embedded environment with Java Virtual Machine.

VII. FROM GATEWAY TO SERVER: WEB SERVICES

Gateway and server exchange data using a defined group of web services that first manage a previous configuration of the server status, including variables and properties, and subsequently manage the data continuous stream.

The first configuration takes place with a request from the gateway that asks for a configuration file, interpret and generates all local components useful to exchange data correctly and to keep synchronization. Immediately after the configuration, the server backend can access the data flow in JSON format, sent from the gateway to the web server using a LAN connection.

The data flow is subsequently managed by the JAVA middleware, which interprets timing and values according to previous configuration, and send all real-time data directly to the frontend web service with few changes, and save preformatted data into the MySQL database. Data to be saved are previously clustered in a time-based manner (e.g. values clustered every five minutes) and then saved into the database.

The time granularity has been decided and set at the first configuration process. Managed data are then made available to web and smart device applications thanks to the RESTful frontend, which offers two types of web services:

- history web service, which collect information from the database and useful to the production manager, to be used outdoor;
- real time web service, which offers real time data, not stored and lost, useful to the smart device hold by the maintenance engineer, to be used indoor.

VIII. INDOOR AND OUTDOOR: THE TWO ROLE-SPECIFIC WEB SERVICES

Server frontend offers two types of web services:

- history web services, to be used by the production manager in order to extract valuable information about the production flow, monitoring past production processes and for being a better decision support system;
- real time web services, useful to the maintenance engineer to inspect production flow and machinery condition monitoring machine alarms and helping him to manage the best machine components setting.

Role-specification of these two web services is immediately clear: the only information each user benefits it's the one he specifically needs, and the information not useful to the specific role it's hidden.

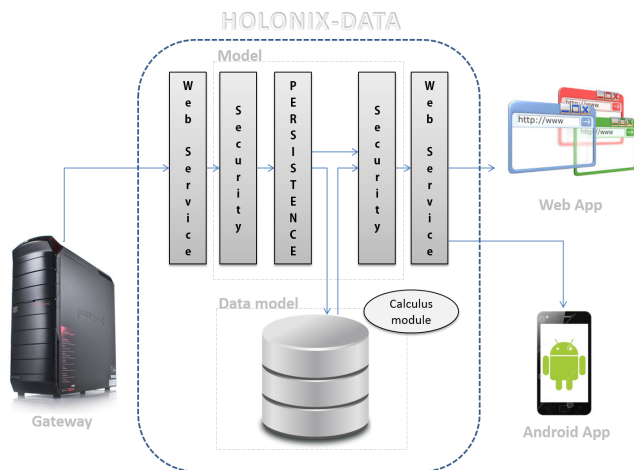


Fig 3. The server architecture model

An HTML5 and JavaScript web application employs the history data, managing them and showing graphic information like charts or tables and useful calculations like sums, counts or aggregate data. The HTML web application is intended to be online, reaching data from the web: that is why it could be defined an “outdoor” application.

An Android application accomplishes the real time data monitoring, taking care about alarms and machine values, showing machine and components status and eventually reporting alarm triggering with some notification sound to the end user, who is intended to be near the machine: that is why this sub-system could be defined “indoor”.

All applications are prototypes, so that a security level has been studied but not applied yet.

IX. RESULTS

The most critical functional requirement to be guaranteed has been the strict time responses of the web application, that was about to reach a maximum of sending through the web connection plus MySQL database record saving of 100 milliseconds.

Tests were clustered in two different types:

- web services response log, while receiving and recording the data flow;
- JavaScript data reading and data showing time.

In all tests, different transducers and alarms types were monitored, automatically simulating some data amount peaks.

A. Web services responses requirements

Web services response log tested the responsiveness of server in receiving, managing and storing data.

The average delay was 5ms, the maximum delay noticed was 241ms and the standard deviation was 0.0233, as shown in “TABLE I”.

TABLE II. WEB SERVICES LOG RESULTS

Web services response log results		
Values number	AVG delay (ms)	MAX delay (ms)
36000	5	241

B. JavaScript response

The JavaScript response test aim was to show how much time was due to pick up values and send them to the web page.

The average delay was around 50ms, the maximum delay noticed was around 80ms on a sample data of 32 values each reading time, as shown in “TABLE II”.

TABLE III. JAVASCRIPT RESPONSE

JavaScript response time results		
Values number	AVG delay (ms)	MAX delay (ms)
-	51	84

C. Stability test

A stability test was made in order to ensure the estimated Cosberg data storage: 5000 values every 24 hours. Test has made for 24 hours recording many more data without additional delay or data loss charged.

D. Database test

During the database inserting process, some complex queries were made, in order to stress the MySQL engine: no timing variation has been detected during data elaboration.

X. CONCLUSIONS AND FUTURE GOALS

Regarding the timing functional needs all performances tests were successful, fully respecting the requirements. The next goal could be find new technologies and environments to further improve performances, like using a no-sql DB, a database considered more fluent in inserting data.

Role-specific splitting of data stream lead to an innovative data classification, which can be considered as an alternative mode to separate big data into various purpose-specific streams. In these terms, next goal should be the creation of a classification model in order to divide large amount of data. In particular authors could approach this issue with some computational intelligence techniques like fuzzy systems, evolutionary algorithms, etc. [9][10][11].

The remote access to machine should become a standard rather than a plus, and every machine should have an integrated Wi-Fi connection that can stream data in a readable way for device applications, in order to fulfill IoT specifications.

Smart devices real-time data monitoring is quite correlated to the previous point, which should assure a good notification system, where “good” means an architecture for example not too heavy for a smart device, which is not necessarily connected to a power source.

REFERENCES

- [1] Apolloni, B., Fiasché, M., Galliani, G., Zizzo, C., Caridakis, G., Siolas, G., Kollias, S., Graña, M., Barrientos, F. and San Jose, S., 2013. “Social and smart: Towards an instance of subconscious social intelligence”.
- [2] Schonsleben, P., Vodika, M., Bunse, K., Ernst, F.O., 2010, The changing concept of sustainability and economic opportunities for energy-intensive industries, *Annals of the CIRP*, 59/1:477-480.
- [3] Bunse, K., Vodicka, M., Schönsleben, P., Brühlhart, M., O. Ernst, F., 2011, Integrating energy efficiency performance in production management: a gap analysis between industrial needs and scientific literature, *Journal of Cleaner Production*, 19/6-7: 667-679.
- [4] Emmanouilidis, C., and Pistofidis, P., (2011), Wireless Condition Monitoring and Embedded Novelty Detection, in *Definitions, Concepts and Scope of Asset Management*, Amadi-Echendu, J.E., Brown, K., Mathew, J., Willet, R., and Mathew, J., pp. 195-238, SPRINGER.
- [5] Lee J., Ni J., Djurdjanovic D., Qiu H. and Liao H, (2006), Intelligent prognostics tools and e-maintenance, *Computers in Industry*, 57 pp. 476–4893. Levrat, E., lung, B., Crespo Marquez, A., (2008). E-maintenance: review and conceptual framework. *Production Planning and Control*, 19(4),408 – 429.
- [6] Muller, A., Crespo Marquez, A., and lung, B., (2008), On the concept of e-maintenance: Review and current research. *Reliability Engineering and System Safety*, no. 93, pp. 1165–1187.
- [7] W.H. Tang and Q.H. Wu, Condition Monitoring and Assessment of Power Transformers Using Computational Intelligence, *POWER SYSTEMS*, 2011, DOI: 10.1007/978-0-85729-052-6
- [8] R. Tetia, K. Jemielniakb, G. O’Donnellc, D. Dornfeldd, Advanced monitoring of machining operations, *CIRP Annals - Manufacturing Technology*, Volume 59, Issue 2, 2010, Pages 717–739, <http://dx.doi.org/10.1016/j.cirp.2010.05.010>
- [9] D. Opresnik, M. Fiasché, M. Taisch, M. Hirsch, “An evolving fuzzy inference system for extraction of rule set for planning a product–service strategy”, *Information Technology and Management*, 2015, DOI 10.1007/s10799-015-0242-4
- [10] M. Fiasché, A.Verma, M. Cuzzola, F.C. Morabito G. Irrera, “Incremental- Adaptive- Knowledge Based- Learning for Informative Rules Extraction in Classification Analysis of aGvHD”, *IFIP Advances in Information and Communication Technology*, Volume 363 AICT, Issue PART 1, 2011, Pages 361-371. DOI: 10.1007/978-3-642-23957-1_41
- [11] M. Fiasché, “A quantum-inspired evolutionary algorithm for optimization numerical problems”. In *LNCS : Vol. 7665*, 2012, *ICONIP 2012*, Part III (pp. 686–693), doi: 10.1007/978-3-642-34487-9_83