Retrofitting a Factory Automation System to Address Market Needs and Societal Changes

Luis E. Gonzalez Moctezuma, Jani Jokinen, Corina Postelnicu, Jose L. Martinez Lastra
Factory Automation Systems and Technology Lab, Tampere University of Technology
Tampere, Finland
luis.gonzalezmoctezuma@tut.fi, jani.t.jokinen@tut.fi, corina.popescu@ieee.org, lastra@ieee.org

Abstract— This paper presents the steps taken to retrofit a real multi robot factory automation testbed to achieve increased flexibility, reconfigurability and awareness of assets including energy-relevant, safety-relevant and quality-relevant parameters. Most changes made can be replicated on similar discrete manufacturing settings, as the leveraging technology is Web Services deploying Service Oriented Architecture. The newly introduced technologies and devices can greatly help to address both needs imposed by the market (e.g. fast product customization) and by the society (e.g. the energy savings desired in the manufacturing domain by 2020).

Keywords— factory automation system, retrofitting, energy savings, flexibility, reconfigurability

I. INTRODUCTION

The approaching of the 2020 horizon highlights the importance of shifting manufacturing industry interests from cost efficiency to high added value (i.e. efficiency in terms of material, energy and waste), to reach the targets set by the European Council in March 2007 (reduction of 20% of the total energy consumption; 20% contribution of renewable energies to total energy production and 20% reduction of greenhouse gases below 1990 emissions).

The increasing demand for greener products translates to the fact that it no longer suffices to produce “good enough” customized products at a reasonably low cost. The inflexibility of Dedicated Manufacturing Systems imposes high modification costs if they must be changed to perform new tasks. To address this problem, Flexible, Reconfigurable and Agile Manufacturing Systems have emerged to leverage the elimination of muda (non-value adding inputs) sought in lean manufacturing [1].

Manufacturing strategies focus mainly on low cost and quality. The notion of quality shifted from being a synonym for guaranteed reliability (at the beginning of the 80s) to variety (at the end of the 80s) and finally to individuality. In the early 90s’ an abrupt increase in product variety caused the competitive advantage to incline towards producers capable not only of creating a steady stream of new products/services, but also take into account the customer’s wishes (see, for instance, the in-house market research program conducted in the 1990s for Airwalk, by Lambesis [2], or the Hyundai Assurance Programme [3]) - to continually add value to existing ones. In agile environments, the pricing of customized goods and services is a function of customer-perceived value: the goal is no longer the lowest cost, but rather the cost that consumers are willing to pay in exchange for the quality they want. As opposed to the unilateral producer-centered customer-responsive companies of lean production (‘choice is driven by the producer, and the responsibility of the client’), agile producer-consumer relationships are interactive (‘choice is driven by the client, and the responsibility of the producer’).

This paper presents the steps taken to retrofit a flexible manufacturing system to make it more responsive to both market needs (i.e. customization and interoperability-related matters ) and societal changes (specifically, the shift towards producing added value with as less resources - energy- as possible). The paper is organized as follows: Section II presents the factory automation testbed in its initial phase. Section III presents the retrofitting steps taken, in terms of technologies and tools utilized. Practical implementations of the retrofitting approach are shown in Section IV. In section V an assessment between the original and the retrofitted industrial system is presented. Finally section VI draws the conclusions of this work.

II. ORIGINAL FACTORY AUTOMATION TESTBED

The initial system is a pallet-based production line consisting of five robotic modular workstations (four assembly stations and one loading/unloading station) previously used in a factory for the assembly of mobile phone components. The system was able to assembly one type of mobile phone at the time.

Pallets travel through the line holding the product under assembly and the components that need to be added. Once product assembly is finished, the finished product is unloaded from the pallet and re-loaded with assembly components.

Each workstation consists of a SCARA robot and a conveyor system. All actuators and sensors are connected through DeviceNet nodes to an OMRON PLC controlling the entire workstation process in a centralized manner.

Workstation controllers communicate with each other through an Ethernet network that runs over coaxial cable, using OMRON FINS protocol [4].

Each pallet of the line contains an RFID tag. RFID readers are placed in the conveyer stoppers to give the workstation
controller information on which pallet is at each stopper. RFID readers are connected via DeviceNet interface that can process the RFID signal to a DeviceNet slave connected to the central PLC.

System communication protocols include DeviceNet (for getting/setting sensor/actuator status), RS-232 serial communication (to send commands to the robot controller), and OMRON FINS/Ethernet (to exchange data between the cells controllers and high level Manufacturing Execution System MES applications) [5].

Each workstation contains acrylic doors to avoid operators to be in contact with equipment in movement. Safety components such as interlock door switches and emergency buttons are connected to safety relays.

Figure 1 illustrates the architecture of the original system for one workstation, the automation components and the used communication protocols.

Figure 1. Original system architecture for one testbed workstation

### III. RETROFITTING THE TESTBED

The main retrofitting goals of the testbed (henceforth denoted as Fastory) were focused on energy, self-recovery capability and possibility to have onsite predictive maintenance.

The original purpose of the testbed was altered, in the sense that assembly operations are simulated by drawing mobile phone components (Frame, Keyboard, Screen). Every drawing operation is equivalent to an assembly operation on physical components in the original setting. All resources in the original setting (e.g. transportation system / robots / pallets / coordination logic, etc) are utilized in a similar way in the new scenario. Each component can be drawn in three different colors. This adds complexity to the system, because of the large amount (>700) of possible product variants. Manipulators, grippers and pallets were redesigned and manufactured for the new purpose of the line.

To boost the number of possibilities available to reconfigure the system, as well as efficiency, responsiveness, information gathering and asset-awareness, it was desired to incorporate into the industrial equipment a processing and communication interface that holds the next features:

- Data processing at device level
- Encapsulation and exposure of the equipment functionality in a homogenous manner
- Transparency in horizontal (device - device) and vertical (device - high level applications) communications.
- Scalability

The testbed hardware was retrofitted with smart remote terminal units (RTU) exhibiting the following characteristics:

- Platform independence
- Plug and play behavior
- Interoperability and extensibility via IT technologies
- Compatibility with major legacy industrial protocols

#### A. Technology selection

As it leverages open, interoperable, scalable and platform independent communications, Web Services (WS) deploying Service Oriented Architecture is the selected technology to achieve the desired requirements. Using WS at device level in the factory floor was recognized to be promising for cross-layer integration and interoperability of applications and equipment. The Device Profile for Web Services (DPWS) stack makes implementation of WS in resource-constrained devices possible, so that they can be discovered and invoked [6].

The commercial product S1000 is an industrial type controller that implements natively the DPWS stack [7]. The S1000 is programmed in Structure Text (ST) language, which is supported by the industrial standard IEC 61131-3. It can interface with sensors and actuators through its IOs, and it has also other (optional) peripherals, e.g. RS232 serial port, Modbus/TCP or analog inputs. An S1000 can communicate with other S1000s and high level applications (irrespective of the platform in which they were developed) through WS messaging. Messaging is carried over IPv6, a communication protocol that offers great scalability properties and is the major candidate to replace the world wide used IPv4 network protocol [8].

#### B. Approach to retrofitting

Each piece of industrial equipment unit in the original setting is composed of three functional layers (Figure 2a):

- The first one is the HW; **physical layer**, which contains the sensors /actuators interacting with the world.
- The second is the **processing layer**, where data is processed and transformed. If the industrial equipment is as simple as an inductive sensor, the data gathered in the physical layer passes without modification towards the processing layer. With more complex industrial equipment e.g. a machine vision system, this layer is in charge of image manipulation via processing algorithms. Usually the customization of this layer is
constrained or null because traditionally the data is processed at the control unit.

The third layer is the interface layer. This layer is a means for the industrial equipment to expose its data or receive commands from an external control unit. For example, for an array of electro valves, a set of digital inputs would be the interface layer. For an energy meter, computed energy parameters could be accessed through a Modbus/TCP interface. An S1000 processing unit (the “smart RTU”) was coupled to the interface layer of the original industrial equipment, thus obtaining:

- **Seamless communication** among pieces of equipment and high level applications.
- **Data processing at device level**, irrespective of its level of complexity [9] (e.g. notifications on whether something is hot/cold coming from temperature sensors already embedding some minimal processing capabilities, such as comparison of sensor values to a threshold to infer more relevant conclusions).

The smart RTU consists of three functional layers (Figure 2b): the interface layer (which must be compatible with the industrial device interface layer); a fully customizable processing layer (responsible with data conversion to information); and the communication layer (exhibiting all above mentioned desired capabilities). The smart RTU can handle one or more industrial components, depending on the capabilities of its interface layer.

![Diagram showing the functional layers of a smart RTU](image)

Industrial components that compose a functional unit of the system (e.g. a conveyor composed by a set of pneumatic actuators, motors and inductive sensors) are grouped together and each group is interfaced through smart RTUs.

The original industrial system uses fieldbus technology, therefore the decision of where to physically place the smart RTU is straightforward: the fieldbus node that sensors/actuators are connected to in the original setting is replaced by the smart RTU.

IV. RETROFITTING FACTORY: PRACTICAL IMPLEMENTATIONS

Six workstations were added to the Fastory line to a total of ten workstations, one loading/unloading station and a static buffer cell (Figure 3). This adds more complexity as the number of scenarios possible in the line is significantly increased, having an impact on pallet routing decisions, line balancing, equipment maintenance and information acquisition systems. The static buffer cell was designed and built to allow removal and retrieval of pallets during the runtime of the process, improving the flexibility of the line because the line production can adapt to different scenarios.

![Image of a factory test bed after retrofitting](image)

All workstations central controllers (OMRON PLCs) were removed. Functional units e.g. robots/conveyors were retrofitted via smart RTUs. This allows having a distributed architecture and therefore distributed control on the system. Conveyors are coupled with the smart RTU through digital IOs. The robot controller is interfaced through digital IOs and RS-232 serial communication. Device functionality is exposed as WS via the DPWS stack natively supported by the smart RTUs.

New pieces of equipment such as pen feeders (used to store and retrieve new pens to the robots when the ink of the current pens finishes) were added to each cell of the original system. The pen feeders simulate the original part feeders that were removed during the retrofitting phase. The functionality of the newly introduced devices is also exposed via WS.

The RFID system was removed and instead of it NFC readers / tags were installed. Each pallet contains a NFC tag. At the entrance of every cell there is an NFC reader. These readers are coupled to the smart RTU that controls the conveyor. The smart RTU communicates with the NFC reader through serial commands. The data of the NFC system is exposed through WS.

The safety system was improved for two of the manufacturing cells, to allow exposure of detailed safety status information as WS. Instead of having a hardware based system, safety PLCs are connected to the cells in order to monitor each of the safety interlock switches and emergency stop buttons. These safety PLCs are connected to a gateway which is monitored by a smart RTU through Modbus/TCP.

Quality inspection is also included in the loading/unloading cell of the Fastory line. All produced products (drawings) are inspected by a smart camera. The smart camera is interfaced through one smart RTU by using Modbus/TCP. In this way quality inspection system can be invoked via WS and quality inspection results exposed in the same manner.

Energy meters have been deployed within each cell, to achieve awareness of power consumption and to profile increase/decrease trends per cell component/line/ product/ cell.
All energy related parameters concerning station components (robots, conveyors, pen feeders) can be processed and published so that the data is further used for predictive maintenance and control purposes. The energy meters run natively the DPWS stack [10].

In the retrofitted Factory, implementations of 6LoWPAN devices are tested. Wireless digital inputs are connected in one of the cells to monitor the safety switches. This allows to expose the information wirelessly also as Web Services, but transported over the 6LoWPAN protocol. Unlike the previously described safety system, which implied modifications done on the hardware, this approach is a plug-in with no interference of safety circuits.

Temperature, humidity and light sensors are connected around the Fastory to monitor wirelessly the industrial facility conditions. This information can be used for industrial and building automation purposes. The data also travels wirelessly through the 6LoWPAN protocol.

To improve the visibility of pallet location and orientation, wireless navigation sensors e.g. accelerometers/gyroscopes are placed in the pallets.

Currently the firmware of the smart RTU supports both IPv4 and IPv6 network protocols. Tests are being conducted to evaluate performance/reliability of IPv6 vs. IPv4 when working under industrial environments.

CAMX standard messages have been deployed in the communications in order to keep consistency in the data structure and semantics [11].

Figure 4 shows the architecture of the retrofitted system. The whole system can be exposed as a set of functional units (robot, conveyor, feeder, etc.). Hierarchies are eliminated since every piece of equipment can be accessed directly in a consistent way by using WS messaging. Third party applications can be attached to the system without disrupting it. Processing of data can be done at device level.

V. ASSESSMENT OF INDUSTRIAL ARCHITECTURES (ORIGINAL VS RETROFITTED)

Several notable differences arise due to retrofitting:

In the original system one type of product can be produced at the time, while the retrofitted system can change the product configuration (customization) during runtime.

In the original system the industrial equipment works as peripherals of a central control unit, where all data is processed. In the retrofitted system data is aggregated and processed into information at device level. Consequently, the coordinating control logic changes: in the original system, sensors have to be scanned constantly to observe changes (passive behavior). In the retrofitted system, events are triggered at device level when special conditions are met (active behavior).

The conceptual visualization of the system also changes. In the original system, the whole set of sensors and actuators have to be coordinated in order to implement the process logic. In the retrofitted system, the sensors and actuators can be grouped in functional units, for example in a robot, in a conveyor or in a feeder. These units are well defined, their functionality is encapsulated and exposed to the rest of the system as WS, and they are independent from other components.

This impacts the modularity of the system. The original line was modular at workstation level. The retrofitted line is modular at equipment level.

The retrofitted system is no longer controlled by a central unit. Distributed units have to interact among themselves and coordinated in a distributed approach.

In the original system, the communications layout is hierarchical. The device layer uses fieldbus technologies, while the controller communications use a high level network like Ethernet. This means that in order to access a device from a high layer application, gateways are needed to pass from one layer to another one. This impacts negatively and hinders the horizontal and vertical integration of the system components. Another drawback is that the data exposed by the industrial components can be in a heterogeneous format, for example, 4-20 mA, digital IOs, Modbus/TCP, RS-232 etc. On the other hand, the retrofitted system eliminates the communication layers and uses just one. This enables devices and applications to communicate seamlessly. Therefore vertical and horizontal integration are transparent. The information is exposed in a homogenous way using SOAP messages. The consistency of the messages is preserved independently from the source. This means that the data coming from a robot, a smart camera, or a single sensor can be treated in the same manner.

The original system uses open automation standards like DeviceNet but encompasses also proprietary protocols like OMRON FINS/Ethernet, while the retrofitted system uses as core technology embedded WS. This leverages the possibility to easily add to the line in future other relevant IT technologies, concepts and tools (e.g. network sniffers, service orchestrators, security mechanisms, parsers, complex event processors, etc).

![Figure 4. Retrofitted system architecture](image-url)
The retrofitted equipment can be monitored remotely by publishing information through internet without major modifications to the system. It suffices to enable a router with Internet connection. This is possible due to the fact that WS are transported over HTTP and can easily travel through standard network appliances. In the original system, a middleware would be needed to poll industrial components through OMRON FINS, and push the polled data towards the monitoring server.

Unlike traditional industrial systems which are protected under the principle of security by obscurity (safety due to system isolation and need of specialized technical knowledge) [12], the retrofitted system, which uses open and well known protocols, must be protected via Defense-in-depth (holding the industrial system behind different layers of network barriers, like firewalls).

Data granularity is improved in the retrofitted system. In the original system it was possible to know the location of a pallet only when it was already in front of an RFID reader. In the retrofitted system the combination of NFC readers and navigation sensors allow to have a better resolution in the pallet location. It is also possible to know relevant events, for example when a pallet is introduced to/retired from the line. Concerning safety data, previously it was possible to know only when the safety system stops a workstation. Now it is possible to know exactly which element triggered the safety event.

There was lack of energy consumption information in the original system. In the retrofitted system, this information can be obtained and used for monitoring and control purposes [13] (e.g. so that products with major energy demand can be produced at times where the energy costs are lower).

Line balancing in the original system was done by tuning process times and sequences. This was possible because the amount of workstations is relative small. In the retrofitted system, the line balancing is based on the independent orchestration of the resources [14].

Table I summarizes the differences between the original and the retrofitted system.

<table>
<thead>
<tr>
<th>DIFFERENCES BETWEEN THE ORIGINAL AND THE RETROFITTED SYSTEM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Production characteristics</strong></td>
</tr>
<tr>
<td>Original: Mass production</td>
</tr>
<tr>
<td>Retrofitted: Reconfigurable</td>
</tr>
<tr>
<td><strong>Horizontal and vertical integration</strong></td>
</tr>
<tr>
<td>Original: Incompatible, Gateways needed</td>
</tr>
<tr>
<td>Retrofitted: Transparent</td>
</tr>
<tr>
<td><strong>Asset awareness</strong></td>
</tr>
<tr>
<td>Original: Location</td>
</tr>
<tr>
<td>Retrofitted: Location + navigation+ events</td>
</tr>
<tr>
<td><strong>Safety information</strong></td>
</tr>
<tr>
<td>Original: No detailed</td>
</tr>
<tr>
<td>Retrofitted: Detailed</td>
</tr>
<tr>
<td><strong>Energy data usage</strong></td>
</tr>
<tr>
<td>Original: None</td>
</tr>
<tr>
<td>Retrofitted: For control and monitoring purposes</td>
</tr>
<tr>
<td><strong>Line balancing</strong></td>
</tr>
<tr>
<td>Original: By tuning times and sequences</td>
</tr>
<tr>
<td>Retrofitted: Orchestration-based</td>
</tr>
<tr>
<td><strong>Communication protocols</strong></td>
</tr>
<tr>
<td>Original: Open/Proprietary</td>
</tr>
<tr>
<td>Retrofitted: Open</td>
</tr>
<tr>
<td><strong>Reusability of IT infrastructure</strong></td>
</tr>
<tr>
<td>Original: Poor</td>
</tr>
<tr>
<td>Retrofitted: High: Software and Hardware</td>
</tr>
<tr>
<td><strong>Plug and play behavior</strong></td>
</tr>
<tr>
<td>Original: Not at all</td>
</tr>
<tr>
<td>Retrofitted: Yes</td>
</tr>
</tbody>
</table>

VI. CONCLUSIONS

Failures in industrial environments can be very costly in time and money. Self-recovery processes can be achieved if the information coming from industrial assets is propagated transparently across the enterprise layers. This information should include energy-related parameters and should be exposed homogeneously.

Legacy industrial equipment can be used with the latest technologies in factory automation if they are retrofitted. This paper discusses the generalization of the functional structure of industrial equipment in an existing testbed, shows how to couple this functionality with a smart RTU and the benefits obtained by doing this. This work was performed on a real test bed, using standard industrial equipment. A generic assessment on the differences between the system before and after the retrofit process is provided.

Production customization, asset awareness including energy-relevant indexes, data granularity, energy-based controls, predictive maintenance are just some of the possible applications of featuring industrial equipment with data processing at device level and a rich communication interface.

ACKNOWLEDGMENTS

This work has partly received funding from the ARTEMIS Second Call 2009 Programme under Grant Agreement number 100223, correspondent to the project shortly entitled eSONIA Embedded Service Oriented Monitoring, Diagnostics and Control: Towards the Asset Aware and Self Recovery Factory.

REFERENCES


