Industrial Automation Services as part of the Cloud: First Experiences

Omid Givehchi
inIT - Institute Industrial IT
OWL University of Applied Sciences
D-32657 Lemgo, Germany
omid.givehchi@hs-owl.de

Jürgen Jasperneite
Fraunhofer IOSB-INA
Application Center Industrial Automation
D-32657 Lemgo, Germany
juergen.jasperneite@iosb-ina.fraunhofer.de

Abstract: Cloud computing has rapidly emerged as an accepted computing paradigm in many areas such as office and enterprise systems worldwide. But still industrial automation domain has slow movement toward this technology due to the automation’s certain requirements which are differed in many aspects from common IT systems. Thus far, existing works related to cloud computing for automation frequently cover higher levels of automation pyramid e.g. enterprise management and process control levels. The main goal of this paper is to draw an image about current state of the research in this field and then introducing first use cases to deliver automation functions as services to the lower levels from the cloud.

1 Introduction

Currently, automation systems are facing fast growing market demands where agility and flexibility in production plants is needed. In addition, over the next years, the fourth industrial revolution will grow based on “intelligent” production. The main focus for Industry 4.0 are smart objects, autonomous products and decision making processes [37] using new technologies from Information Technology domain.

Cloud computing, as a new trend from the IT area, might become an enabler for these future automation systems, because it has recently influenced many areas such as office and enterprise systems. Cloud computing has rapidly emerged as an accepted computing paradigm in many enterprises worldwide due to its flexibility and many other advantages [2].

The main goal of this paper is to introduce the first experiences of applying cloud computing solutions for industrial automation systems. Our work is based on already proposed architectures for future automation systems. Since these architecture proposals are limited to very abstract models for future systems, this work aims to analyze related works in this research area and offer the first implementations based on current research gaps.

The remainder of this paper is organized as follows. After a short introduction, state of the art for cloud computing technology will be provided based on its architecture and important features and types in section 2. New architectures for industrial automation will be described in section 3 by starting from a motivation for the change in current architecture and the derivation of future architectures based on the corresponding needs. In section 4, existing approaches that apply cloud computing in industrial automation are discussed. Based on these findings a summary of existing research gaps which need to be addressed in the future is provided. In section 5, first implementations to offer cloud services for the automation is described with details. The last section will conclude the work and briefly address future work in this area.

2 Cloud Computing

Computing models during their history have been directly affected by other available technologies in that specific period of time. Before introducing the PC in 1980s, centralized computing using Mainframes was the only available solution for computing which was similar to the idea of centralized electricity distribution opined by Parkhill [15]. Personal Computing was the first local computing solution to the problem of availability which was based on the idea of decentralized computing. Consequently, it caused to widespread computing and IT everywhere, because of low cost resources and simplicity. Afterwards, computer networks and especially Internet has got more popularity and bandwidth as well as stronger hosts started to grow. On the other hand, by emerging “Everything as a Service (XaaS)” [36] concept in the computing as an approach to pack and deliver computing resources as services via a central service provider, the IT world motivated for a new computing
Cloud computing has different definitions and understandings from different perspectives and applications. The National Institute of Standards and Technology (NIST) defined cloud computing as “a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.” [25]. From the scientific viewpoint, the main goal of cloud computing is to provide on-demand computing services with high scalability and availability in a distributed environment with minimum complexity for the service consumers.

Cloud computing architecture can be divided into several layers: the hardware layer, the infrastructure layer, the platform layer and the application layer [28] as shown in Figure 1.

Hardware layer handles the physical resources of the cloud, including physical hardware, network devices and power systems. Typical issues at the hardware layer include hardware configuration, fault-tolerance, traffic management, and power management. Infrastructure layer is also known as virtualization layer. This layer partitions the hardware and provides a pool of computing resources and disk storage. Platform layer is mainly covers operating systems and application frameworks depending on each specific platform. This layer tries to minimize the development efforts by providing development platform to the developers as a service without installing any software or framework on their computers. Application layer offers the cloud applications to the end users as a service. These applications can be automatically scaled with high performance inside this layer with lower maintenance costs comparing with traditional applications [39].

Cloud computing is based on a service-driven model. In this model, hardware and software resources will be delivered as services on-demand. These fundamental cloud services are categorized into three models:

**IaaS** provisions infrastructural resources such as virtual machines on-demand. It is the most essential cloud service model. IaaS providers or cloud owners e.g. Amazon EC2, GoGrid and Flexiscale offer their resources to the users with least complexity using this service model.

**PaaS** provides platform layer resources e.g. software development frameworks and deployment components. Software developers employ these services to develop and deploy applications with minimum installation and preparation of resources. Google App Engine, Microsoft Windows Azure and Force.com are examples of PaaS providers.

**SaaS** offers on-demand cloud applications to the users through network. This service offers complete complexity abstraction for the users. They do not need to deal with preparing required hardware and software resources and application is accessible through a standard interface e.g. web browsers. Examples of SaaS providers include Microsoft Office 365, Google Calendar and SAP Business ByDesign.

Besides, based on applications and architecture of clouds, they can be divided into four different types.

**Public Cloud** offer cloud-based applications and services to the general public via the Internet. Numerous organizations and users can use the resources from an infrastructure at the same time. Benefits of this cloud type include less investment for users to install and maintain infrastructures at their location and outsource these operations to the providers. However, public clouds limit the control over data privacy and security settings
which is undesirable in many areas. This model is also known as external cloud.

**Private Cloud** provides an exclusive use of cloud resources inside a company or organization. It can be implemented and managed by the owner or by an external provider. A private cloud offers the highest degree of control over cloud resources, privacy and security comparing with other cloud types. However, its costs are higher than other cloud types. This type is also known as internal cloud.

**Hybrid Clouds** is a combination of public and private clouds. In this type some services are offered from private part while others are provided from public cloud. This combination addresses the limitations for both parts that offer more flexibility comparing with other cloud types. Though there is a complexity to distinguish different private and public clouds at the development time, especially for security and access control initialization of the system.

**Community Cloud** is a shared cloud infrastructure between several users from a specific community in order to share common services, security and compliance among them. The costs of this implementation is more than public cloud, but less than private type. Since this infrastructure is shared between a number of companies, the costs will be divided accordingly.

### 3 Industrial Automation Architectures

During recent years, industrial automation witnessed the new challenges in different areas. Among with increasing market demands for agile and smart manufacturing, new trends in this domain has appeared. By embedding new functionalities inside field devices, e.g. IO devices or sensors and actuators, they become more intelligent now. Similarly communication between these devices has been developed significantly by improving communication standards and protocols, e.g. Profinet [17] and OPC [7] in the lower levels which makes devices more connected via standard communications.

These trends result to define new automation architectures with different to the current hierarchical automation pyramid. Among these newly proposed architectures, Vogel-Heuser et. al. [35] introduced a new architecture as global information architecture for industrial automation. It consists of two main layers presented with two cones which are placed between business and technical processes. The lower layer represents the migration of field and control layers in traditional pyramid including devices and functionalities, whereas the upper layer represents the process control and management layers on top of the control layer in the automation pyramid.

![Figure 2: Global Information Architecture with use of Cloud Computing](image)

At the end of this proposal authors described the new model as a motivation and introduction for a change in current automation architecture. They promised an improvement in engineering life-cycle and more flexible operations. However, no general solution has been described for implementing the proposed idea. Based on this, cloud computing technology has been applied in [29] as a solution to implement this newly proposed architecture for automation. The architecture is shown in Figure 2. Special automation functions and services can be offered directly as SaaS from the IT-cloud as well as Automation-Cloud. Alternatively, a PaaS is used as a automation platform to deliver specific needs for integration, e.g. process logs and plug and play parameters.

As shown in Figure 2, automation cloud (AT-Cloud) is offered to provide functions and services in lower levels
and IT-Cloud hosts applications and services in upper levels of the automation. Integration between these two infrastructures are provided by an Information Model. Considering common cloud computing features and architecture, the Information Model could be replaced with direct service interactions which displayed in Figure 3 with solid lines. As a result of this migration, along with the definition of standard services, it is possible to replace IT and Automation clouds with a single cloud illustrated in Figure 3 as a cyber-physical system (CPS). The integration between the cloud and real devices will be enabled by encapsulating services and functions inside delivery standards. In our proposed architecture this is introduced as Everything-as-a-Service (XaaS) for automation.

As shown in Figure 3, the critical control and field levels still exist in a traditional way based on the well-known automation pyramid. However, the upper levels have been changed. Since they are basically providing non-physical functions and services, it is possible to migrate them into the cloud. Nevertheless, these functions and services are still categorized with respect to the well-known levels of the pyramid.

![Figure 3: Decomposition of the automation hierarchy by CPS with distributed services](image)

The control level is divided into two parts. The first part is the physical control level which includes common PLCs close to the technical process to be able to provide the highest performance for critical control loops. The second part of the control level migrated to the cloud. It offers control functions directly to the lower levels from the cloud with a reduced performance.

## 4 Related Work

Basically, existing works related to cloud computing in automation could be categorized based on their focus on each individual automation level. Most of these works aim at migrating functions and services from the common hierarchical automation architecture to a flat architecture.

### 4.1 Enterprise Management and Manufacturing Execution Level

Among the works related to the higher levels of the automation especially enterprise management level, Xu et al. [38] discussed some of the essential features for cloud computing with regard to apply them in manufacturing management systems. Gilart-Iglesias in [9] proposed a service model for delivering industrial machinery as a service to incorporate them easily during production process in order to facilitate self-management and proactive management of the business logic for which it is responsible.

### 4.2 Process Control Level

For the process control level, several research projects especially related to Service Oriented Architectures (SOA) have been done already. Related to these works, Delsing et al. [5] proposed an approach to migrate from legacy industrial systems to the next generation of SOA-based automation systems as a part of SOCRADES [3]
project following the ISA95 standard [30]. The main idea for this work is to migrate automation functions in
different levels to a cloud of services and to connect these services together via a service bus which provided by
SOA. As a result of this migration, services can be ran and accessed through the new service infrastructure with
more flexibility and scalability and integration between them will be improved [19]. Gerach et al. [8] proposed
a private cloud model to host and deliver SIMATIC PCS7 as a generic Distributed Control System (DCS).
Furthermore, Web-oriented Automation System (WOAS) [21] project aims to research a new architecture for
automation systems based on web and cloud technologies.

### 4.3 Control and Field Level

Related to the lower levels of automation, i.e. control and field levels, number of existing works are mainly
limited due to the tough requirements of these levels. These levels basically consist of physical devices and
functions. There are several solutions related to the sensor cloud which are described in [12] for general
purposes and IT applications. A first use case for engineering applications as a service has been investigated
in [10]. The main focus was to deliver engineering applications from the cloud to the user without having
installation and maintenance efforts.

### 4.4 Summary

After analyzing the state of the art, a lack of projects and research works in the area of cloud computing for
automation was noticed, especially lower levels of automation are not sufficiently addressed. The main reason
for this could be the leakage of information about this buzzword in automation industry which is known as a
conservative domain against new technologies [11]. The summary of existing works is shown in Table 1. As
viewed, some of the requirements, e.g. real-time and security are not sufficiently addressed and remain as open
research questions for the future.

### 5 Implementations

In this section, first implementations of cloud services for industrial automation will be described. The main
goal of these implementations is to address the possibility of applying cloud solutions specially in lower levels
of automation. Since this work is in the beginning phases, a simple use case has been used to demonstrate the
concept. A comprehensive implementation and evaluation for this work is still in progress and will be discussed
in future works. Figure 4 shows an overview of implemented cloud components which will be further described
in detail.

#### 5.1 Control-as-a-Service

Controllers play an important role in automation systems. In distributed automation systems, each individual
controller or PLC device provides independent control functions for each separate module of devices. These

devices are needed to be connected to the PLC by means of electrical signals and I/O devices conventionally. With growth of Ethernet-based industrial networks such as Profinet and Ethercat, it is possible to control the field devices directly from the network. This shapes the main idea of delivering control functions as services to the field floor. For this reason, Microsoft Windows Azure\(^1\) is picked as the main cloud platform provider for the implementation.

Typically, Azure is known as a public cloud solution from Microsoft which is accessible from Internet. However, public cloud solutions are not interested by automation applications due to security and reliability concerns. Therefore, private cloud solutions from Azure is selected for this implementation. In this case, cloud provider will be installed and provisioned on a private and local server connected to the local network of the plant. On the top of the host operating system, a hypervisor with automatic provisioning feature is installed which enables system to deliver virtual machines automatically as they are demanded. Here each virtual machine can be considered as individual controller or any automation’s service provider. The communication between these virtual machines and components of the system will be handled by a Service Bus which has compatibility with different Service Oriented Architecture standards for integration. The architecture of this system is showed in figure 5.

A Profinet protocol stack from KW-Software enables the virtual machine to communicate with field devices. A control application is developed to control a production module in LeMgo smart Factory\(^2\) (LMF). Devices installed on this module are able to connect to the Profinet network through an I/O controller. The control software is developed in order to receive production start and duration signals as input and sending control functions to the actors on the module. Furthermore, a SOAP web client is developed in ANSI C and added to the mentioned control code. As it shown in figure 4, this web client connects to the orchestrator web service via service bus and invokes required control parameters. In our use case, client asks for production start time

---

1. www.windowsazure.com
2. www.smartfactory-owl.de
and duration and sets this parameters in control execution code meanwhile. Since this client is developed with ANSI C, it is possible to migrate its code to the PLC device and call it as a function block in the future. This would be assumed as a new trend while current IEC 61131-based tools does not support web services functions. First results of this implementation generally meet basic application requirements since this application does not have high real-time requirements. Because operating system and control software are running on top of the cloud hypervisor, a guaranteed determinism is not achievable for this controller. Currently, generic cloud hypervisors have no support for real-time applications. However, there are some available solutions with real-time virtualization support such as real-time XEN [22] which can be used as cloud hypervisor in the future works.

5.2 Services Orchestration

Generally, controllers should be programmed to run the technical process in industrial systems. Among with increasing number of devices and services, manual orchestration of these services becomes a complex and time consuming task. Recently, there are some ongoing projects to develop service orchestration solutions for automation such as SOCRADES [3], SIRENA [16] and Grafchart [33]. These works aim to implement service orchestration applications using Service Oriented Architecture and based on automation needs. Meanwhile, high efforts in IT domain have been done in the business process orchestration area. The main focus of these works is to orchestrate business processes using intelligent approaches such as semantic definitions of services. BPEL (Business Process Execution Language) and OWL-S (Web Ontology Language for Services) as two matured orchestration languages are compared in [27]. The results of comparisons show that OWL-S is more appropriate for dynamic environments and BPEL is offered for controlled workflows. BPEL has also capability to orchestrate semantic web services by generating ontologies and import them to the orchestration process model [26]. In our work, a static BPEL workflow introduced to demonstrate a possible use case of offering service orchestration from the cloud.

As shown in figure 4, a BPEL engine called Apache Orchestration Director Engine (ODE) [1] is running on Virtual Machine 3. This engine consists of a web service and an add-on installed on top of it that supports BPEL process models deployment. This means that it is sufficient to design process workflow and transfer it to the engine for automatic deployment without additional efforts. A sample BPEL process workflow is illustrated in figure 6. This BPEL process is basically designed for initializing the parameters for the controller running on Virtual Machine 2. As mentioned in last subsection, this controller is implemented to control the production module’s devices in the LMF. A web client is integrated with control code which asks SOAP values from ODE web service periodically.

As soon as receiving request by orchestration engine, it runs process sequences in parallel to invoke required values from other web services. In our work, three external web services are invoked to get required values and calculate total module 6 production time. Based on environment temperature’s value received from weather web service, heating time for production can be determined in first sequence. Second sequence reads available bottle numbers in production queue and initializes a suitable production duration for total number of bottles. Third sequence is designed to show an actual vertical integration with management level. In this sequence, heating duration for each bottle will be initialized by invoking feedback from management and customers provided by a enterprise web service. After running all sequences, orchestration engine will calculate the control parameters and return them to the controller’s web client inside a SOAP response envelope.

Typically, static workflows in current state of the art automation systems are programming with accepted standards such as IEC 61131. However when it comes to orchestration between different entities in different levels, some weaknesses could be realized for current standards. With current implementations of IEC 61131, there is no support for some Service Oriented Architecture key technologies such as SOAP and RESTful web services. On the other hand, despite IEC 61131 has extremely support for static workflows, still there is no dynamic orchestration features supported by it. These two features would be assumed as advantages for orchestration solutions from IT world since dynamic orchestration of automation processes is one of the key requirements for future industry systems. Nevertheless, a concrete comparison between these approaches should be investigated for future works. Lastly, cloud would be a proper solution to deliver these orchestration services.
5.3 Engineering Applications-as-a-Service

The high variety of engineering applications in different levels of automation leads to complex tasks of installation and maintenance for plants. Using cloud service delivery model, these applications can be encapsulated inside the services and delivered to the users through the network independent from a particular platform. As it mentioned in the cloud architecture, Software as a Service (SaaS) is the top best service that can be offered to users from the cloud layers of architecture. This software would be considered as industrial engineering applications, which are used by users to configure the different devices in the plant. For this reason, Citrix XenApp [14] as a application virtualization solution is installed on Virtual Machine 1 as shown in figure 4.

Consequently, engineering applications should be installed and configured on the server once and could be ran by several users on different platforms through the network numerously. In this case, computing and processing are performed centrally in the cloud and users do not require to deal with complexity of installation and maintenance of the applications on their local devices. An evaluation of this approach has been done in [10] based on measured performance of the PCWORX engineering tool application delivered from XenApp.

6 Conclusion

With increasing usage of information technologies such as the Internet of Things, Service Oriented Architectures and mobile computing in industrial automation, there is a further need for a platform to provide computing resources, information integration and a data repository for the connected devices (things). Cloud computing could be a possible solution for this reason. On the other hand, this technology could be assumed as a solution for implementation of newly proposed architectures for automation. This would be possible with migration of current automation functionalities to the cloud. However, application requirements determine potential for each function for moving to the cloud or staying in the floor as a physical device. In this work, some of these functions are introduced as the first cloud-based services for automation.

In future works, control level must be further investigated in order to address reliability and real-time issues. To satisfy industrial companies for migrating to cloud-based systems, new approaches should be applied based on vendor independent standards to ensure their interoperability with special consideration of security issues.
Acknowledgments

This work was partly funded by the EU FP7 STREP project IoT@Work under grand number ICT-257367 and the German Federal Ministry of Education and Research (BMBF) within the Leading-Edge Cluster "Intelligent Technical Systems OstWestfalenLippe" (it’s OWL) and managed by the Project Management Agency Karlsruhe (PTKA). The authors are responsible for the contents of this publication.

References


