

Object-Oriented Internet Cloud Interoperability

Mariusz Postół¹[0000–0002–9669–0565] and Piotr Szymczak¹[0000–0001–6790–3878]

Institute of Information Technology, Lodz University of Technology, Łódź, Poland
<mailto:mariusz.postol@p.lodz.pl>
<http://www.it.p.lodz.pl>

Abstract. Optimization of the industrial processes requires further research on the integration of machine-centric systems with human-centric cloud-based services in the context of new emerging disciplines, namely Industry 4.0 and Industrial Internet of Things. This research aims at working out a new generic architecture and deployment scenario applicable to this integration. The reactive interoperability relationship of the interconnected nodes is proposed to deal with the network traffic propagation asymmetry or assets' mobility. Described solution based on the OPC Unified Architecture international standard relaxes issues related to the real-time multi-vendor environment. The discussion addressing the generic architecture concludes that the embedded gateway software part best suits all requirements. To promote separation of concerns and reusability, the proposed architecture of the embedded gateway has been implemented as a composable part of a selected OPC UA PubSub framework.

The proposals are backed by proof of concept reference implementations confirming the possibility to integrate selected cloud services with the cyber-physical system interconnected as one whole atop of the OPC UA by applying the proposed architecture and deployment scenario. It is contrary to interconnecting cloud services with a selected OPC UA server limiting the PubSub role to data export only.

Keywords: Industry 4.0 · Internet of Things · Object-Oriented Internet · Cloud Computing · Industrial communication · Reactive Networking (RxNetworking) · Machine to Machine Communication · OPC Unified Architecture · Azure

1 Introduction

All the time, the Information and Communication Technology is providing society with a vast variety of new distributed applications aimed at micro and macro optimization of the industrial processes. Obviously, the design foundation of this kind of application has to focus primarily on communication technologies. Based on the role humans take while using these applications they can be grouped as follows:

- **human-centric** - information origin or ultimate information destination is an operator,
- **machine-centric** - information creation, consumption, networking, and processing are achieved entirely without human interaction.

A typical **human-centric** approach is a web-service supporting, for example, a web user interface (UI) to monitor conditions and manage millions of devices and their data in a typical cloud-based IoT approach. In this case, it is characteristic that any uncertainty and necessity to make a decision can be relaxed by human interaction. Coordination of robots behavior in a work-cell (automation islands) is a **machine-centric** example. In this case, any human interaction must be recognized as impractical or even impossible. This interconnection scenario requires the machine to machine communication (M2M) demanding multi-vendor devices integration.

From the M2M communication concept, a broader idea of a smart factory can be derived. In this M2M deployment approach, the mentioned robots are only executive assets of an integrated supervisory control system responsible for macro optimization of an industrial process composed as one whole. Deployment of the smart factory concept requires a hybrid solution and interconnection of the mentioned above heterogeneous environments. This approach is called the fourth industrial revolution and was coined as Industry 4.0. It is worth stressing that machines - or more general assets - interconnection is not enough, and additionally, assets interoperability has to be expected for the deployment of this concept. In this case, multi-vendor integration makes communication standardization especially important, namely, it is required that the payload of the message is standardized to be factored on the data-gathering site and consumed on the ultimate destination site.

Highly-distributed solutions used to control real-time process aggregating islands of automation (e.g. virtual power plants producing renewable energy) additionally must leverage public communication infrastructure, namely the Internet. Internet is a demanding environment for highly distributed process control applications designed atop the M2M communication paradigm because

- it is a globally shareable and can be also used by malicious users
- it offers only non-deterministic communication making integration of islands of automation designed against the real-time requirements a difficult task

Today both obstacles can be overcome, and as examples, we have bank account remote control and voice over IP in daily use. The first application must be fine-tuned in the context of data security, and the second is very sensitive in regard to time constraints. Similar approaches could be applied to adopt the well known in process control industry concepts:

- Human Machine Interface (HMI)
- Supervisory Control and Data Acquisition (SCADA)
- Distributed Control Systems (DCS)

A detailed examination of these solutions is far beyond the scope of this article. It is only worth stressing that, by design, all of them are designed on the foundation of interactive communication. Interactive communication is based on a data polling foundation. In this case, the application must follow the interactive behavioral model, because it actively polls the data source for more information by pulling data from a sequence that represents the process state in time. The application is active in the data retrieval process - it controls the pace of the retrieval by sending the requests at its convenience. Such a polling pattern is similar to visiting the books shop and checking out a book. After you are done with the book, you pay another visit to check out another one. If the book is not available you must wait, but you may read what you selected. The client/server archetype is well suited for the mentioned above applications.

After dynamically attaching a new island of automation the control application (responsible for the data pulling) must be reconfigured for this interoperability scenario. In other words, in this case, the interactive communication relationship cannot be directly applied because the control application must be informed on how to pull data from a new source. As a result, a plug and produce scenario [3] cannot be seamlessly applied. A similar drawback must be overcome if for security reasons suitable protection methods have been applied to make network traffic propagation asymmetric. It is accomplished using intermediary devices, for example, firewalls, to enforce traffic selective availability based on predetermined security rules against unauthorized access.

Going further, we shall assume that islands of automation are mobile, e.g. autonomous cars passing a supervisory controlled service area. In this case, the behavior of the interconnected assets is particularly important concerning the environment in which they must interact. This way we have entered the Internet of Things domain of Internet-based applications.

If we must bother with the network traffic propagation asymmetry or mobility of the asset network attachment-points the reactive relationship could relax the problems encountered while the interactive approach is applied. In this case, the sessionless publisher-subscriber communication relationship is a typical pattern to implement the abstract reactive interoperability paradigm. The sessionless relationship is a message distribution scenario where senders of messages, called publishers, do not send them directly to specific receivers, called subscribers, but instead, categorize the published messages into topics without knowledge about which subscribers if any, there may be. Similarly, subscribers express interest in one or more topics and only receive messages that are of interest, without knowledge about which publishers, if any, there are. In this scenario, the publishers and subscribers are loosely coupled i.e they are decoupled in time, space and synchronization [1].

If the **machine-centric** interoperability - making up islands of automation - must be monitored and/or controlled by a supervisory system cloud computing concept may be recognized as a beneficial solution to replace or expand the mentioned above applications, i.e. HMI, SCADA, DCS, etc. Cloud computing is a method to provide a requested functionality as a set of services. There are many

examples that cloud computing is useful to reduce costs and increase robustness. It is also valuable in case the process data must be exposed to many stakeholders. Following this idea and offering control systems as a service, there is required a mechanism created on the service concept and supporting abstraction and virtualization - two main pillars of the cloud computing paradigm. In the cloud computing concept, virtualization is recognized as the possibility to share the services by many users, and abstraction hides implementation details.

Deployment of the hybrid solution providing interoperability of the **machine-centric** Cyber-Physical System (CPS) and **human-centric** cloud-based front-end can be implemented applying the following scenarios: **direct** or **gateway** based interconnection (Sect. 2.1).

By design, the **direct** approach requires that the cloud has to be compliant with the interoperability standard the CPS is built upon - it becomes a consistent part of the CPS. Data models, roles, and responsibility differences of both solutions make this approach impractical or even impossible to be applied in typical cases. A more detailed description is covered by the Sect. 2.

This article addresses further research on the integration of the multi-vendor **machine-centric** CPS designed atop of M2M communication and emerging cloud computing as a **human-centric** front-end in the context of the Industry 4.0 (I4.0) and Industrial Internet of Things (IIoT) disciplines. For this integration, a new architecture is proposed to support the reactive relationship of communicating parties. To support the multi-vendor environment OPC Unified Architecture [2] interoperability standard has been selected. Prototyping addresses Microsoft Azure Cloud as an example. The proposals are backed by proof of concept reference implementations - the outcome has been just published on GitHub as the open-source (MIT licensed) [4]. The proposed solutions have been harmonized with the more general concept called the Object-Oriented Internet [4].

The main goal of this article is to provide proof that:

1. reactive interoperability M2M communication based on the OPC UA standard can be implemented as a powerful standalone library without dependency on the Client/Server session-oriented archetype
2. Azure interoperability can be implemented as an external part employing out-of-band communication without dependency on the OPC UA implementation
3. the proposed generic architecture allows that the gateway functionality is implemented as composable part at run-time part - no programming required

The remainder of this paper is structured as follows. Sect. 2 presents the proposed open and reusable software model. It promotes a reactive interoperability pattern and a generic approach to establishing interoperability-context. A reference implementation of this archetype is described in Sect. 3. The most important findings and future work are summarized in Sect. 4.

2 Sensors to Cloud Field Level Connectivity

2.1 Architecture

subsection

2.2 Azure Main Technology Features

subsection

subsubsection

subsubsection

subsubsection

2.3 OOI Main Technology Features

subsection

3 Azure - Object-Oriented Internet Interoperability Implementation

section

4 Conclusion

section

References

1. Eugster, P.T., Felber, P.A., Guerraoui, R., Kermarrec, A.M.: The many faces of publish/subscribe. *ACM Computing Surveys* **35**(2), 114–131 (2003). <https://doi.org/10.1145/857076.857078>
2. González, I., Calderón, A., Figueiredo, J., Sousa, J.: A literature survey on open platform communications (opc) applied to advanced industrial environments. *Electronics* **8**, 510 (05 2019). <https://doi.org/10.3390/electronics8050510>
3. Pfrommer, J., Stogl, D., Aleksandrov, K., Navarro, S.E., Hein, B., Beyerer, J.: Plug and produce by modelling skills and service-oriented orchestration of reconfigurable manufacturing systems. *At-Automatisierungstechnik* **63**(10), 790–800 (2015). <https://doi.org/10.1515/auto-2014-1157>
4. Postól, M.: Object oriented internet; azure gateway implementation 1.0 (Dec 2020). <https://doi.org/10.5281/zenodo.4361640>, <https://github.com/mpostol/OPC-UA-OOI>