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. It is characteristic that, in this case, 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, it is essential that any human interaction is 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 concept of a smart factory can be derived. In this concept, the mentioned robots are only executive assets of an integrated supervisory control system responsible for macro optimization of an industrial process composed into one whole. Deployment of the smart factory concept requires a hybrid solution and interoperability of the mentioned above heterogeneous environments. This approach is called the fourth industrial revolution and 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, muli-vendor integration makes communication standardization especially important, namely it is required that the payload of the message is standardized to be factored on the 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 environment 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 on time relationships. Similar approaches could be applied to adopt the well known in process control industry concepts:

- Human Machine Interface (HNI)
- 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 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 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 archetype is a typical pattern to implement the abstract reactive interoperability paradigm. The sessionless archetype 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 \cite{RefWorks:doc:5c44e246e4b0591b15ea9e59}.

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 systems and **human-centric** cloud-based frond-end can be implemented applying the following scenarios:

- **direct interconnection** - cloud-based dedicated communication services allow to attache it to the cyber-physical system making up a consistent M2M communication network using an in-bound protocol stack
- **gateway based interconnection** - typical build-in communication services allows to attache the cloud computing to the cyber-physical system using an out-of-bound protocol stack

By design, the direct approach requires that the cloud has to be compliant with the interoperability standard the cyber-physical system is built around - it becomes a consistent part of the cyber-physical system. Data models, roles, and responsibility differences of both solutions make this approach impractical and imposable to be applied in typical cases. A more detailed description is covered by the section `Azure to Sensors (A2S) connectivity deployment`.

This article addresses further research on the integration of the cyber-physical systems in the context of new emerging disciplines, i.e. Industry 4.0 (I4.0) and the Internet of Things (IoT). The new architecture is proposed for integration of the multi-vendor **machine-centric** cyber-physical system designed atop of M2M reactive communication and emerging cloud computing as a **human-centric** frond-end. To support the multi-vendor environment OPC Unified Architecture interoperability standard has been selected. The proposals are backed by proof of concept reference implementations. Prototyping addresses Microsoft Azure Cloud as an example. The prototyping outcome has been just published on GitHub as the open-source (MIT licensed). The proposed solutions have been harmonized with the more general concept called the Object-Oriented Internet.

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 composable at run-time - no programming required

Sect. `Azure Main Technology Features` analyzes data presentation user interface, available native communication services, and data/metadata model offered by the Microsoft Azure. The discussion covered by this section is the foundation for selecting services utilized to expose process data and suitable protocol stack to support interconnection. In Sect. `Object-Oriented Internet Main Technology Features` the discussion focuses on the generic architecture that is to be used as a foundation for further decisions addressing the systematic design of the interoperability of the cyber-physical systems and cloud-based front-end. Sect. `Azure to Sensors (A2S) connectivity deployment (field level connectivity)` 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. `Gateway implementation`. The most important findings and future work are summarized in Sect. `Conclusions`.

Consider to add

Scope

What we must do to prove the goal have been achieved. Extent or range of development, view, outlook, application, operation, effectiveness, etc.

Related work

Any information about available reusable deliverables related to this work.