

Data supply chain in Industrial Internet

Andrea Buda, Kary Främling, Jukka Borgman,
Manik Madhikermi, Saeed Mirzaeifar
Aalto University - School of Science - Department of
Computer Science
Espoo, Finland
andrea.buda@aalto.fi

Sylvain Kubler
Luxembourg University,
Centre for Security, Reliability & Trust line
Luxembourg
sylvain.kubler@uni.lu

Abstract— The Industrial Internet promises to radically change and improve many industry's daily business activities, from simple data collection and processing to context-driven, intelligent and pro-active support of workers' everyday tasks and life. The present paper first provides insight into a typical industrial internet application architecture, then it highlights one fundamental arising contradiction: “Who owns the data is often not capable of analyzing it”. This statement is explained by imaging a visionary data supply chain that would realize some of the Industrial Internet promises. To concretely implement such a system, recent standards published by The Open Group are presented, where we highlight the characteristics that make them suitable for Industrial Internet applications. Finally, we discuss comparable solutions and concludes with new business use cases.

Keywords—Industrial Internet; Data analysis; Web Protocol; IoT; Open Messaging Interface.

I. INTRODUCTION

The term “*Industrial Internet*” has been coined by General Electric [1]. The term is relatively recent (November 2012), but it has already catalyzed the attention of many leading organizations. In March 2014, the Industrial Internet Consortium¹ was formed to bring together Industry, Government and Academia in a global non-profit partnership to accelerate and promote the growth of the underlying technologies, use cases and best practices.

The Industrial Internet is not a new technology, it is simply a “catch-all” term for existing technologies and disciplines applied in industrial contexts; such as:

- Machine-to-machine (M2M) communication;
- Internet of things (IoT) and Cyber-Physical Systems (CPS)
- Machine Learning and Data Mining
- Big Data

These disciplines interact and cooperate together in the collection of machine and human generated data (in digital format) during business operation, which is then analyzed (in real-time or off-line) and used to adjust the behavior of given equipment.

The expected innovation should come naturally from the application of the right set of existing technologies to right use cases. It is interesting to notice that the whole initiative is not

waiting or hoping for new technologies, instead there is some sort of realization about “how much more can be achieved with current one”.

Another important point to highlight is that advanced networking is not the main focus of the Industrial Internet (even though the name is a slightly misleading). The main focus is the systematic exploitation of machine generated data to optimize operations. The benefits from the marriage of machines and analytics are multiple and significant. General Electric estimates that the technical innovations of the Industrial Internet could find direct applications in sectors accounting for more than \$32.3 trillion in economic activity. As the global economy grows, potential applications of the Industrial Internet will expand as well. By 2025 it could be applicable to \$82 trillion of output or approximately one half of the global economy, adding a sizable \$10-15 trillion to the global GDP [1].

Currently the Industrial Internet consortium is essentially driven by United States enterprises, meanwhile in Europe similar initiatives have different names. *Industry 4.0* is a project in the high-tech strategy of the German government that promotes the computerization of the manufacturing industry [2], while the upcoming European Union Framework Programme for Research and Innovation (Horizon 2020 - running from 2014 to 2020), has a dedicated track named Factories of the Future (FoF) [3]. As mentioned above, all these initiatives aim at combining well developed disciplines, taking the most suitable technology for the task at hand and promote them through real world industrial use cases.

This paper first describes the common architecture adopted in typical Industrial Internet applications. The selected model comes from the authors' direct experience and is essentially the same one found in the IoT and M2M literature. Then it highlights the shortcoming of current approaches to Industrial Internet applications, which have direct impacts on what kind of data analysis can be performed. At a more practical level, the present paper provides a scenario in which machine data must be shared among suppliers in order to provide advanced analytics to the final customer. To support this data supply chain, recent standards published in October 2014 by The Open Group² are considered as foundation of the communication layer. These standards are the Open Messaging Interface O-MI [4] and Open Data Format (O-DF) [5]. Section 4 discusses how

¹ <http://www.iiconsortium.org>

² <http://www.opengroup.org/>

such standards mitigate the identified shortcoming, the conclusions follow.

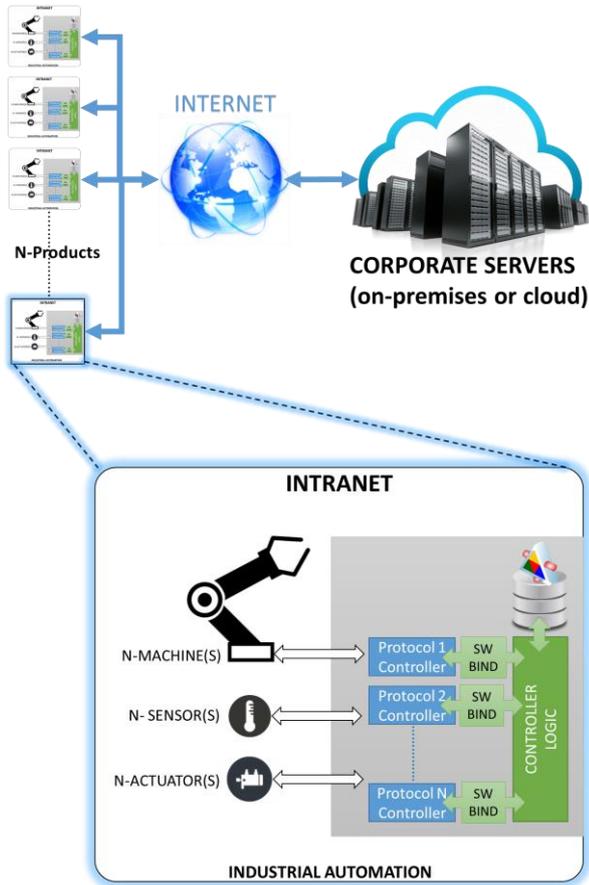


Fig. 1. Typical Industrial Internet Architecture

II. INDUSTRIAL INTERNET TYPICAL ARCHITECTURE AND SHORTCOMING OF THE CURRENT APPLICATIONS

Figure 1 depicts a simplified architecture diagram of a typical Industrial Internet application, which consists of three main layers [6]:

1. An internal high trust intranet. In this layer we found, machines, PLCs, drives, sensors and actuators connected together through a multitude of industrial automation networking protocols. This network is hierarchical, where every machine, machine subsystem, and subsystem component has its own local network and controller (see Figure 2). For example modern servo motor drives, are at the bottom of this hierarchy but they can still be considered as complex systems with their own set of sensors and smart adaptive controller. Just like classes in object-oriented programming, at every level all this complexity is encapsulated and hidden while few simple functions are exposed.
2. On the edge of this intranet, there is a gateway to expose and transport data to the external world (the public internet). In general data from the internal network is aggregated in a relational or NoSQL databases. Even

though it is possible to connect directly to these databases using HTTP or HTTPS or even plain TCP/IP, most implementation prefer to expose a single endpoint with few defined operation. This endpoint is usually implemented using SOAP [7], web services, or REST API [8]. This paper focuses on this category of protocols, specifically on O-MI that has some significant advantages compared to these custom defined endpoints.

3. The receivers of this data are usually some corporate servers running either on the cloud or on-premises. The architecture described in [6] referred this layer as “application domain”. In these servers data is analyzed and some KPIs (key performance indicators) are derived. The nature of the analysis depends on the application. In general Industrial Internet applications aim at optimizing the operational behavior of a given product/system or a fleet of products/systems.

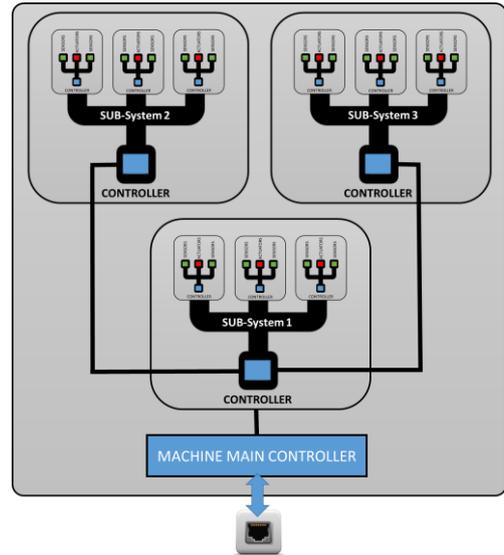


Fig. 2. Modern Machine is a system of systems, each sub-system is usually provided by external specialized supplier

A. Shortcoming of the current applications

The application and architecture described in the previous paragraph has few shortcomings, and in particular one major contradiction that must be addressed before realizing the full potential of the Industrial Internet paradigm: *Who owns the data is often not capable of analyzing it*; “not capable” does not mean that they do not know how to create SQL analytic reports or running data mining algorithms. What is often missing is the specific domain knowledge to understand the data. The next section provides a concrete example to better highlight this statement.

B. Future Scenarios – The Industrial Internet Data Supply Chain

Let us consider the following scenario: a small medium enterprise (SME) produces multiple sheet metal parts for a variety of customers. This enterprise is highly automated and owns a state-of-art sheet metal punching machine, which has a

number of sub-system resembling as illustrated in Figure 2. Some of those sub-systems are supplied to the punching machine manufacturer by specialized companies; for example, once the subsystem is the punching head smart drive and integrated motor. This subsystem has its own set of sensors, controllers and embedded software. Finally the machine is integrated in the entire production system by an industrial automation consulting company directly hired by the owner of the factory.

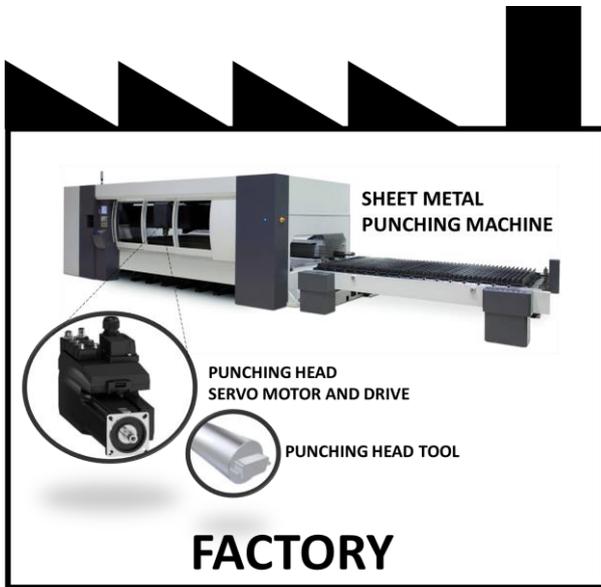


Fig. 3. Instantiated example of the architecture depicted in figure 1

The scenario includes three main players:

1. *Factory owner* (working together with the industrial automation consulting company);
2. *Sheet metal punching machine manufacturer*;
3. *Smart-drive manufacturer*;

According to the Industrial Internet vision, all those actors can benefit from data generated during daily operations. For example:

1. *The factory owner* can optimize scheduling and production to increase throughput and decrease costs;
2. *The sheet metal punching machine and smart-drive manufacturer* can learn how their equipment operate in the field. For example, they can correlate alarms and errors with production condition and machine settings. This correlation can be used to adjust some setting or completely update the firmware of their equipment.

However there are few obstacles in the realization of these benefits. The owner (together with the industrial automation consulting company) ultimately decides which data will leave his company. Very often if there are no immediate gains, security policies dictate avoid opening any data to the external world. Interesting enough, if the factory owner decides to let the punching machine manufacturer access his equipment, the machine manufacturer has now the power to decide whenever the smart drive manufacturer can access the data regarding his

equipment. Unfortunately this situation is only partially solvable by technology. What is also needed, is the realization that the cooperation for a more flexible data exchange can generate benefits for everybody. In particular, it is possible to leverage on specific knowledge necessary to analyze certain datasets. For example, let us consider the following use cases:

Punching tool wearing: If the punching tool is not sharp enough, the sheet metal will get stuck in the punching head. In case of highly automated line without human supervision, this error might stop production for an entire work shift. Because a variety of sheet metal thickness and materials are punched (which wear the tool in different way), simply counting the number of punches cannot accurately predict whenever a tool needs sharpening. However, the machine producer has the intuition that when the tool is dull the smart drive will draw current in a different way than when the tool is sharp. At this point, he explains this possibility to factory owner who, in turn, will provide the access to the machine and detailed information about the tool he is using. A study is conducted by storing the current drawing time series evolution. The time series, together with contextual production condition information, is then forwarded to smart drive manufacturer who will analyze it and extract a model predicting the tool wearing situations. The newly discover knowledge is coded and shipped with the new release of the punching machine controller software.

The data-supply chain described above has direct benefits for all players. The smart drive producer can directly bill his analysis to the punching machine manufacturer, strengthening their relationship. In turn the machine manufacturer has new features in his machine that will make it possible to strengthen the relationship with his customer and hopefully attract some new one. In order to realize this visionary scenario, many points, both technological and organizational must be addressed. The following section presents a step forward towards this vision by briefly introducing O-MI and O-DF standard specifications.

III. O-MI AND O-DF STANDARDS OVERVIEW

The Open Group IoT work group has a pretty clear and ambitious vision: Whereas the Web uses the HTTP protocol for transmitting HTML-formatted information which are rendered in the browser for human consumption, the IoT will use O-MI for transmitting O-DF payloads which will be mainly consumed by information systems. The initial ideas and requirements for these protocols emerged out of the PROMISE EU FP6 project, where real-life industrial applications required the collection and management of product instance-level information for many domains involving heavy and personal vehicles, household equipment, etc. Information such as sensor readings, alarms, manufacturing, disassembly, and supply chain-events, and other information related to the entire product lifecycle needed to be exchanged between products and systems of different organizations. The main goal was to find solutions that would enable the communication between intelligent production in their middle of life (MoL), and backend information systems. These functions are often referred as called Closed-Loop

Product Lifecycle Management (CL₂M) [9] [10] [11]. Based on the needs of those real-life applications, key functional requirements were identified and extended to provide a more general solution for large-scale scalable IoT systems (see Table 1). As no existing standards could be identified that would fulfill those requirements without extensive modification or extensions, PROMISE partners started working on specifications that would fulfill the gap, which ultimately lead to the development [12][13] and standardization of O-MI and O-DF.

TABLE I. KEY FUNCTIONAL REQUIREMENTS

ID	DESCRIPTION
1	Generic enough for use in any kind of IoT systems, independently of the application domain.
2	Support for “synchronous” messaging such as immediate read and write operations.
3	Handling mobility and intermittent network connectivity, i.e. support for asynchronous messaging capabilities that imply for instance message persistence, time-to-live, etc.
4	Ability to establish secure, two-way communication even in the presence of firewalls without requiring firewall re-configuration
5	Possibility to create ad hoc, loosely-coupled, time-limited information flows “on the fly”, i.e. without programming.
6	Peer-to-peer communication possibility for all devices, i.e. client and server functionality can be implemented for any device, depending on available processing power, network connectivity, etc.
7	Context-dependent discovery of instances, instance-related services and meta-data about them.
8	Support for context- and domain-specific ontologies.
9	Queries by regular expressions for retrieving information about more than one instance and more than one kind of information.
10	Historical queries, i.e. retrieving values between two points in time.

Open data format (O-DF)

O-DF is specified as an extensible XML Schema. It is structured as a hierarchy with an “Objects” element as its top element. The “Objects” element can contain any number of “Object” sub-elements. It is intentionally defined in a similar way as data structures in object-oriented programming. Figure 4 depicts an example of an O-DF message (schema location and other informative header has been removed to improve readability). The text in blue represents the actual O-DF message while the header in red describes the O-MI operation. The “Object” elements can have an arbitrary number of InfoItems as well “Object” sub-elements. In this example, the Object *SmartFridge_SN5622334411* has only one InfoItem named *Temperature*. InfoItems can contain the following sub-elements:

- **Metadata (optional):** Sub-element that provides meta-data information about the InfoItem, such as value type, units and other similar information.
- **Value:** Value for the InfoItem, possibly with an optional timestamps.

Every Object has a compulsory sub-element called “id” that identifies the Object. The “id” should preferably be globally unique or at least unique for the specific application, domain,

or network of the involved organizations (e.g. Serial Number of the appliance).

```

<omi:omiEnvelope>
  <omi:response>
    <omi:result msgformat="odf">
      <omi:returnCode> 200 </omi:returnCode >
      <omi:requestId>REQ654534</omi:requestId>
      <omi:msg>
        <Objects>
          <Object>
            <id>SmartFridge_SN5622334411</id>
            <metadata>FIU20WH MK2</metadata>
            <InfoItem name="Temperature">
              <metadata>
                <units>Celsius</units>
                <writable type="xs:boolean">false</writable>
                <readable type="xs:boolean">true</readable>
              </metadata>
              <value type="xs:double"
                unixTime="5453563">3.5</value>
            </InfoItem>
          </Object>
        </Objects>
      </omi:msg>
    </omi:result>
  </omi:response>
</omi:omiEnvelope>

```

Fig. 4. O-MI (in black) and O-DF (in gray) example message.

The “id” plays a crucial role in Product Lifecycle Management applications. As information about a given equipment are often spread over several information systems and organizations, the “id” represents the “where” condition of a hypothetical SQL query, that can be used to obtain additional information about the specific object.

In Logistics and Product Lifecycle Management, the univocal identification of objects is a well know problem that has been researched for quite a while [14] [15]. New research is addressing unsolved issues, however an extensive review of the subject is out of the scope of this paper.

The standard also specifies how O-DF is used as a query language for specific parts of the Object tree (Value if an InfoItem, Metadata about it, a branch of the tree or even the entire tree), as well as for publishing the Object tree using URLs in as resources in a RESTful [13] manner.

Open Messaging Interface (O-MI)

O-MI standard is a web protocol designed to expose and transport physical products data (including sensor, actuators and any other machine information) from local intranets to any other destination connected to the Internet. As depicted in Figure 1, O-MI sits on top existing Application level protocol, which means that O-MI operations can be transported using any suitable “lower level” protocol. In this regard O-MI is similar to SOAP and different from REST that implies HTTP.

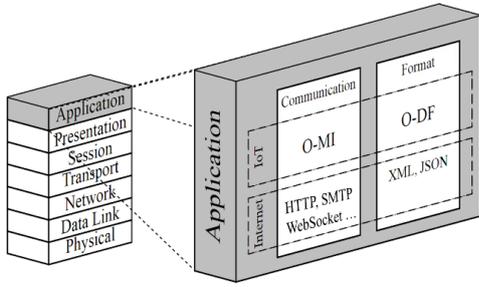


Fig. 5. O-MI and O-DF in the ISO/OSI stack

The standard supports 3 main operations:

1. **READ:** There are two types of read i) for immediate retrieval of data; for placing subscription to an O-MI node including three additional parameters:
TTL (time-to-live): For how long (in seconds) the subscription is valid.
INTERVAL: Sets the rate at which the data will sample in seconds. Setting the interval to -1 is equal to sample a given value(s) whenever it changes.
CALLBACK ADDRESS (optional): This parameter specifies the URL to which the node should respond to. Figure 6 and 7 illustrate the subscription sequence diagram without and with the callback address parameter;
2. **WRITE:** It is a simple update operation of a given InfoItem.
3. **CANCEL:** Cancelling subscriptions before they expire.

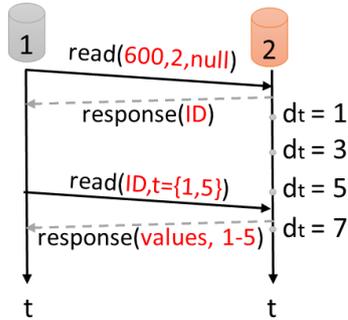


Fig. 6. O-MI READ read(ttl,interval, callback address) sequence diagram. Callback Address NOT specified.

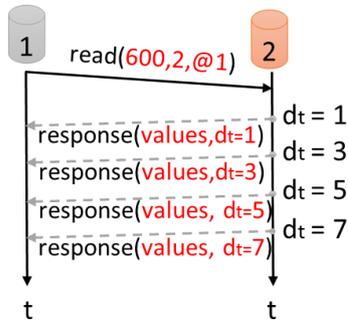


Fig. 7. O-MI READ read(ttl,interval, callback address) sequence diagram. Callback Address specified.

Other important characteristics that have been proven useful, especially in real-world implementations are:

- Payload agnostic: Even though the preferred payload is O-DF (XML or JSON formatted), in reality any payload could be transported (CSV, HTML, proprietary file formats);
- Possibility to perform a machine/product initiated communication, while the receiving node can place a new O-MI request over the same connection. This possibility, also known as piggy-backing is described in [16], is very useful for addressing nodes that are located behind firewalls or NATs;
- The standard [9] also specifies how O-DF is used as a query language for specific parts of the Object tree (Value if an InfoItem, MetaData about it, a branch of the tree or even the entire tree), as well as for publishing the Object tree using URLs in a RESTful manner. The fact that O-DF covers all steps of the publish-discover-query-reply cycle is an advantage for software implementation because the same O-DF structure can be re-used in all steps.

IV. DISCUSSION

The possibility to create ad hoc, time-limited information flows (requirement number 5 of Table 1) by specifying for how long (TTL) and at which sampling rate (INTERVAL) the data needs to be received, is the cornerstone of O-MI, which makes it particularly suitable for IoT and Industrial Internet applications. If we consider the “punching tool wearing” scenario described in Section 2, the entire production line and all its subsystems can generate a tremendous amount of data, enabling endless data analysis possibilities. However, the punching machine manufacturer and the industrial automation consulting company will use only a small subset of all available data. They cannot predict what kind of analysis one might want to perform in the future, and it impossible to log everything. There is no “big data” solution that can support the logging at the maximum sampling rate of every subsystem of a complex machine, and even if such solution would exist, it simply does not make any sense to do it, mainly because the value of data collection is realized only when the data is analyzed and new knowledge is derived.

Therefore, once it is clear how certain data can be exploited, it is necessary to have a system capable of retrieving specific data, at a given interval and for a sufficient amount of time, which would allow to perform the planned analysis. This is exactly what O-MI has been designed for, and what differentiate it from existing protocols.

Related Protocols Landscape

A detailed comparison between O-MI and similar messaging protocols or libraries is out of the scope of this paper. However, it is important to mention comparable solutions, and position O-MI in IoT landscape. The IoT landscape is large and heterogeneous, from cloud-based systems to embedded software and M2M communication. It is nearly impossible to give a single and unified picture of the overall landscape of IoT standards due to its heterogeneity and complexity [17].

Nonetheless, in order to illustrate the main layers and solutions that shape this landscape, we have used an illustration (Figure 8) from [18] that shows many of the most relevant existing solutions/standards for the IoT. Xu et al. [19] summarize the current state-of-the-art of IoT applications and technologies used in industry, focusing on the link, transport and communication layers (as depicted in Fig. 8).

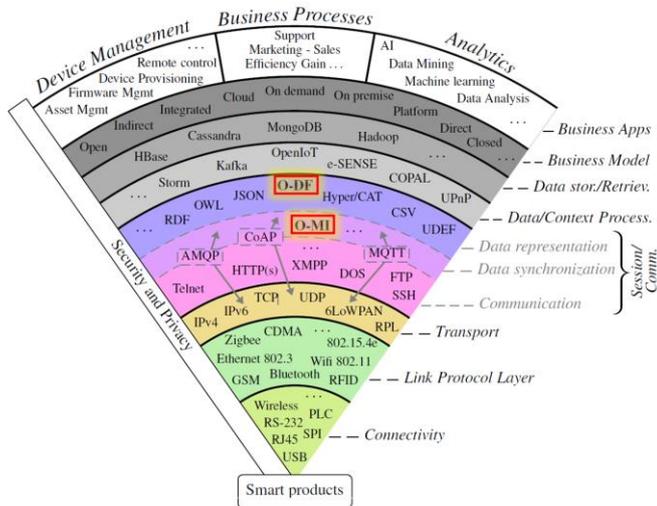


Fig. 8. IoT standards'/protocols' landscape: adapted from [18]

In general O-MI can be associated with SOAP web-services or a RESTful API. However SOAP and REST are essentially an evolution of RPC-XML (Remote Procedure Call over XML), therefore there is no predefined behavior definition for a given operation/call, while O-MI support only a limited set of operations. Clearly this can be seen as limitation, however a considerable amount of time has been spent by Open Group to define the smallest set of operations which would enable the vision described in Section 2 and 3. The standardization of these simple operations should increase interoperability and hopefully stop the proliferation of custom web APIs.

O-MI can also be associated with some existing message-oriented middleware, especially when it comes to READ with deferred retrieval (aka Subscriptions). Messaging solutions can be divided in two categories:

- **Messaging standards:** Like AMQP (Advanced Message Queuing Protocol), which is mostly designed for enabling enterprise messaging, or MQTT (Message Queue Telemetry Transport) a light weight messaging protocol, designed to run on devices with constrained resources.
- **Messaging APIs:** Like JMS (Java Message Service) or ZeroMQ. Which are not standardized protocol, therefore are implementation depended. In other words the same library must be used on both ends to enable communication.

In addition JMS, AMQP and MQTT rely on high-availability brokers, while O-MI operates more in a peer-to-peer fashion as much as ZeroMQ. A fair comparison among all these solutions is not an easy task, and the results will most likely depend on the application requirements.

V. CONCLUSIONS

This paper presents a visionary data supply chain for future Industrial Internet, where main obstacles (organizational and technological) that need to be overcome to fully realize all the benefits of an Industrial Internet paradigm are set out. Recent IoT standards published by The Open Group that play a major role in enabling data supply chain are briefly introduced. The possibility to create ad hoc, time-limited information flows by specifying for how long and at which sampling rate we need to receive the data, is the cornerstone of O-MI and future Industrial Internet applications. Future work will continue to apply and investigate the protocol in industrial use cases. Trying to improve it, from cooperation and feedback of both the Open Group and the Industrial Internet Consortium members.

ACKNOWLEDGMENT

This research was conducted in the Future Industrial Services (FutIS) research program, managed by the Finnish Metals and Engineering Competence Cluster (FIMECC).

This work was funded by the Finnish Funding Agency for Technology and Innovation (TEKES) and the European Commission through the Linked Design Project (FoF-ICT-2011.7.4: Digital factories: Manufacturing design and product lifecycle management, <http://www.linkeddesign.eu/>). The authors wish to acknowledge their gratitude and appreciation to the rest of the project partners for their contributions during the development of various ideas and concepts presented in this paper.

REFERENCES

- [1] P.C.Evans, M.Anunziata, Industrial Internet: Pushing the Boundaries of Minds and Machines, November 2012, Available online at http://www.ge.com/docs/chapters/Industrial_Internet.pdf
- [2] German Federal Ministry of Education and Research, Project of the Future: Industry 4.0 - Available online at <http://www.bmbf.de/en/19955.php>
- [3] European Commission, Factories of the Future PPP: towards competitive EU manufacturing, Available online at http://ec.europa.eu/research/press/2013/pdf/ppp/fof_factsheet.pdf
- [4] Open Messaging Interface (O-MI), an Open Group Internet of Things (IoT) Standard, US ISBN 1-937218-60-7, Oct. 2014, Available online at <https://www2.opengroup.org/ogsys/catalog/C14B>
- [5] Open Data Format (O-DF), an Open Group Internet of Things (IoT) Standard, US ISBN 1-937218-59-1, Oct. 2014, Available online at <https://www2.opengroup.org/ogsys/catalog/C14A>
- [6] H. Chen, J. Xueqin, and L. Heng, A brief introduction to IoT gateway, ICCTA 2011 Proceedings, (2011): 610-613.
- [7] SOAP Version 1.2 specification. Available online at <http://www.w3.org/TR/soap/>
- [8] R. T. Fielding, "Architectural styles and the design of network-based software architectures", PhD thesis: University of California, 2000.
- [9] D. Kiritsis, A. Bufardi, and P. Xirouchakis, Research issues on product lifecycle management and information tracking using smart embedded systems. Advanced Engineering Informatics., 2003, Vol. 17, 189-202.
- [10] D. Kiritsis and A. Rolstadås, PROMISE-A closed-loop product lifecycle management approach. In IFIP 5.7 Advances in Production Management Systems: Modelling and Implementing the Integrated Enterprise, 2005 (NIST Publishing: USA).

- [11] H. B. Jun, J. H. Shin, D. Kiritsis, and P. Xirouchakis. System architecture for closed-loop PLM. *International Journal of Computer Integrated Manufacturing*, 20(7):684–698, 2007
- [12] D. Kiritsis, 2010. Closed-loop PLM for Intelligent Products in the era of the internet of things. *Computer Aided Design* 43 (5), 479–501.
- [13] K. Främling, J. Holmström, J. Loukkola, J. Nyman, A. Kaustell, Sustainable PLM through Intelligent Products. *Engineering Applications of Artificial Intelligence*, Volume 26, 2012. pp. 789-799. <http://dx.doi.org/10.1016/j.engappai.2012.08.012>.
- [14] E. Huvio, J. Grönvall, K. Främling, Tracking and tracing parcels using a distributed computing approach, in: Proc. 14th Annual conference for Nordic researchers in logistics, 2002, 29–43.
- [15] K. Främling, M. Harrison, J. Brusey, J. Petrow, Requirements on unique identifiers for managing product lifecycle information: comparison of alternative approaches, *International Journal of Computer Integrated Manufacturing*, 20 (7) (2007) 715–726.
- [16] S. Kubler, K. Främling, A. Buda, A standardized approach to deal with firewall and mobility policies in the IoT, *Pervasive and Mobile Computing*, Available online 7 October 2014, ISSN 1574-1192, <http://dx.doi.org/10.1016/j.pmcj.2014.09.005>.
- [17] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey”, *Computer Networks*, vol. 54, no. 15, 2010, pp. 2787–2805.
- [18] O. Vermesan, and P. Friess, *Internet of Things – From Research and Innovation to Market Deployment*, River Publishers, 2014.
- [19] L.D. Xu, W. He, S. Li, Internet of Things in Industries: A Survey, *IEEE Transactions on industrial informatics*, Vol. 10, no. 4, november 2014