



Technical Interoperability for Machine Connectivity on the Shop Floor

Downloaded from: <https://research.chalmers.se>, 2019-05-11 19:29 UTC

Citation for the original published paper (version of record):

Åkerman, M., Stahre, J., Engström, U. et al (2018)
Technical Interoperability for Machine Connectivity on the Shop Floor
Technologies, 6(3)
<http://dx.doi.org/10.3390/technologies6030057>

N.B. When citing this work, cite the original published paper.



Article

Technical Interoperability for Machine Connectivity on the Shop Floor

Magnus Åkerman ^{1,*} , Johan Stahre ¹, Ulrika Engström ², Ola Angelsmark ³, Daniel McGillivray ², Torgny Holmberg ³, Maja Barring ¹, Camilla Lundgren ¹, Martin Friis ⁴ and Åsa Fast-Berglund ¹

¹ Industrial and Materials Science, Chalmers University of Technology, 412 58 Gothenburg, Sweden; johan.stahre@chalmers.se (J.S.); maja.barring@chalmers.se (M.B.); camilla.lundgren@chalmers.se (C.L.); asa.fasth@chalmers.se (Å.F.-B.)

² Ericsson Research, 417 56 Gothenburg, Sweden; ulrika.engstrom@ericsson.com (U.E.); daniel.mcgillivray@ericsson.com (D.M.)

³ Ericsson Research, 223 62 Lund, Sweden; ola.angelsmark@ericsson.com (O.A.); torgny.holmberg@ericsson.com (T.H.)

⁴ Manufacturing Development Centre, SKF (Svenska Kullagerfabriken AB), 415 50 Gothenburg, Sweden; martin.friis@skf.com

* Correspondence: magnus.akerman@chalmers.se; Tel.: +46-(0)729-636518

Received: 2 May 2018; Accepted: 19 June 2018; Published: 22 June 2018



Abstract: This paper presents a generic technical solution that can increase Industry 4.0 maturity by collecting data from sensors and control systems on the shop floor. Within the research project “5G-Enabled Manufacturing”, an LTE (Long-Term Evolution) network with 5G technologies was deployed on the shop floor to enable fast and scalable connectivity. This network was used to connect a grinding machine to a remote private cloud where data was stored and streamed to a data analytics center. This enabled visibility and transparency of the production data, which is the basis for Industry 4.0 and smart manufacturing. The solution is described with a focus on high-level communication technologies above wireless communication standards. These technologies are discussed regarding technical interoperability, focusing on the system layout, communication standards, and open systems. From the discussion, it can be derived that generic solutions such as this are possible, but manufacturing end-users must expand and further internalize knowledge of future information and communication technologies to reduce their dependency on equipment and technology providers.

Keywords: interoperability; IoT; connectivity; Industry 4.0; smart manufacturing

1. Introduction

The next generation cellular wireless technologies, 5G, will enable a scalable all-in-one connectivity platform that supports connectivity of ubiquitous equipment, enabling a fourth industrial revolution (Industry 4.0). The technology only partially exists today, as requirements are being refined and formalized into the open 3GPP (3G Partnership Project) international standard that is set for the near future [1]. However, connectivity infrastructure is only one piece of the puzzle in the goal of achieving highly flexible fully interconnected manufacturing systems. As suggested by Vernadat [2], the ability for systems to interconnect can be divided into technical, semantic, and organizational levels, which are incorporated in the holistic concept of interoperability. Technical interoperability includes connectivity but also deals with syntactical and architectural aspects. Interoperability has been an important part of manufacturing system design since software systems were first introduced to aid

resource management during the Computer Integrated Manufacturing (CIM) era during the 1990s. Since then there has been a rapid development of information technologies, e.g., Cloud Computing, Big Data, Internet of Things (IoT), Internet of Services (IoS), and Cyber-Physical Systems (CPS) [3–7]. Such technologies need to be further adopted by the industry in order to build smart manufacturing systems, which emphasizes the importance of interoperability [8,9].

The work presented has been conducted within the research project “5G-Enabled Manufacturing” (5GEM). The project is a collaboration between a large manufacturing company (SKF, Gothenburg, Sweden), a large telecommunication system provider (Ericsson, Gothenburg, Sweden), and academia (Chalmers University of Technology, Gothenburg, Sweden). The project investigates the general question of how 5G networks can be utilized in a production system. The key question is how the manufacturing industry can enhance the speed and quality of the application of new communication technology to improve manufacturing performance. The aim of this paper is to present challenges and solutions regarding technical interoperability in an industrial setting. A real-world case of connecting a grinding machine at SKF to a remote private cloud is described and presented in detail, focusing on technologies beyond fast and scalable wireless connectivity. The results regarding high-level technical interoperability are discussed, divided into the three areas: System layout, communication standards, and open systems, to increase Industry 4.0 maturity.

2. Background

2.1. Industrial Digitization and Industry 4.0 Maturity

A key issue for manufacturing enterprises is to ensure that the application of new technologies adds value to their core business. The goal of Industry 4.0 is to create cyber-physical production systems (CPPS) from which data-driven autonomous systems can emerge. The Industry 4.0 Maturity Index model [10] is a way to measure how close an enterprise is to reaching this goal. The model sets several prerequisites needed for an industrial system to take advantage of new capabilities. To reach these different steps, many things need to be in place. Figure 1 shows a high-level list of what technological solutions, or paradigms, that can be loosely connected with each step. The first step, computerization, is to have a digital system as opposed to a purely mechanical one, a development that triggered the third industrial revolution during the 1950s. The second step is to connect the various digital systems. Academic discussion around the complexity of such interconnections intensified during the 1980s, during the Computer Integrated Manufacturing (CIM) era, which leads to multiple frameworks of enterprise integration and interoperability [2,11].

The third step is termed visibility, i.e., having a digital model of the factory. Data from machines and sensors need to be collected and stored continuously so that the current state of the production system is always known and based on facts. Collecting all the relevant data in a complex manufacturing environment requires an efficient and scalable information system. A solution is to utilize shared resource pooling and Cloud Computing technologies [12]. According to the National Institute of Standards and Technology (NIST), cloud computing “is 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” [3].

Industry 4.0 maturity index four is called transparency, which means to make sense of the digital models to understand the current situation. A central aspect of this is to use algorithms for data analysis in Big Data applications. A popular definition of Big Data includes the four important V's of data, drawn from an IDC (International Data Corporation) report 2011 [13]. “Big data technologies describe a new generation of technologies and architectures, designed to economically extract value from very large volumes of a wide variety of data, by enabling high-velocity capture, discovery, and/or analysis.”

In the fifth step, predictive capacity, real-time data and analyzed information needs are linked, aggregated, simulated, etc., into useful information which can be used for decision-making. This aligns

with a certain aspect of IoT that focuses on a layered architecture and the use of middleware [5]. Encapsulating functionality in well-defined services enables and simplifies data sharing between new and legacy systems with IoS [6,14].

The sixth and last step in the Industry 4.0 Maturity Index model is called adaptability. This is the end goal of both Industry 4.0 and IoT, which is also referred to as smart manufacturing or smart industry [15,16]. A Cyber-Physical Production System (CPPS) is an adaptable system with the capability to sense its environment and dynamically react to it. Since this reaction needs to function at different time scales, depending on which part of the system needs to adapt, there is a need for programming and modeling languages with such support [7].

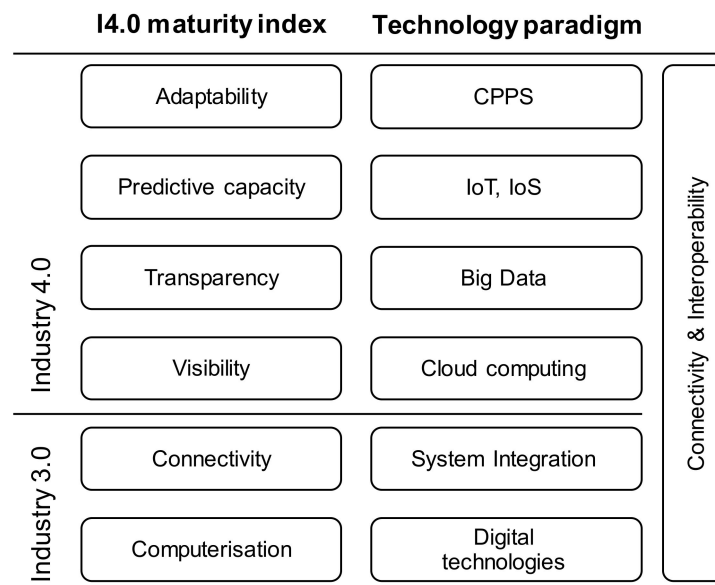


Figure 1. A high-level illustration of what technological solutions, or paradigms, that can be loosely connected with each step in the Industry 4.0 maturity index model [10].

2.2. Technical Interoperability and Enabling Technologies

Interoperability issues relate to technical, semantic, or organizational aspects [2]. Technical interoperability aspects regard the data exchange between different information technologies at the physical and syntactical level. When considering data for CPS in manufacturing, it is important to differentiate between configuration data and run-time data [7]. Configuration data is generated during the design of the system and describes the physical parts of the system. Run-time data is generated during the operational phase and describes the status of the manufacturing process. Optimal synergies are achieved when these data types can share the same models, which require usage of industrial standards for both design and operations. The automation engineering standard Automation ML (AML) and communication standard OPC Unified Architecture (OPC UA) are examples of standards that can be semantically linked [17].

Another core consideration when interconnecting systems is layout, meaning the way in which each entity relates to others in terms of computation and decision making. The classical control system is hierarchical and deterministic, operating with centralized decision making. In such systems, data only flows vertically between child and parent nodes. The IoT paradigm assumes a decentralized approach where data can flow horizontally. Such systems are event-driven and non-deterministic where a more complex system emerges, which needs new methods to be managed [7]. A practical approach to achieve CPS in manufacturing is to create a semi-hierarchical approach with emphasis on modularization. An example is a solution by Wang and Haghghi [18], promoting a holonic (modularized) system where each Holon is controlled by a software agent (autonomous unit).

Approaches to the layout problem, developed outside of the manufacturing domain, include various platforms and middleware for IoT applications that have gained traction lately. IoT platforms, also called frameworks or middleware, are systems that simplify the integration of things (e.g., sensors, machines, and equipment) through decoupling to enable IoT applications. There are three types of IoT systems: Service-based, cloud-based, and actor-based [19]. Service-based systems are centralized heavy-weight generic platforms often deployed in a cloud environment [20]. Cloud-based systems are application specific services often tied to a specific product. Actor-based middleware systems are flexible and scalable since they support deployment in different layers. Examples of actor-based middleware are the Ptolemy framework, Node-RED, and Calvin [21–23]. Actors are reusable function blocks that can easily be deployed on different hardware. Some platforms also allow migration of actors to where and when they are needed, as long as the system supports the needed requirements [24,25]. This allows computations closer to the network edge, thus allowing for a more decentralized system, known as edge-networks or fog computing [26].

IoT systems are natural aggregators of data, often including data analytics capabilities for Big Data applications. When data volumes grow the system requires technologies that can distribute computing and storage capabilities and send a large amount of data between different nodes. There are several open-source projects that can enable these systems, such as Hadoop, Spark, and Kafka [27–29] for distributed computing and data streaming. For data storage, the NoSQL databases, which are more flexible than the traditional relational databases, is a key technology [4]. The term NoSQL, which should be read as “not only SQL”, incorporates characteristics like non-relational, distributed, open-source, and horizontally scalable [30].

Frameworks for CPS and IoT often include a middle layer that connects objects and applications through services [31–33]. This is called a Service Oriented Architecture (SOA), whose principle is to connect business functions (services) between providers and consumers by using service locators or service brokers [34]. A definition of a service, therefore, needs to account for business and technical perspectives as well as the consumer and provider perspectives. However, from the technical perspective, a service is an encapsulation of functionality [35]. SOA is often used synonymously with Web-services, but this is only one implementation. Web service API's (Application Programming Interface) often comply with the Representational State Transfer (REST) style that implies stateless interactions and a hierarchical resource representation [36]. SOA is also supported by the above-mentioned communication protocol for industrial automation, OPC UA. New communication protocols have also been developed to support ubiquitous and sometimes limited hardware, such as Message Queuing Telemetry Transport (MQTT) and Constrained Application Protocol (CoAP) [37,38].

Interconnection of all the ubiquitous equipment that is assumed in the IoT paradigm requires a fast, robust, and scalable connectivity platform. The next generation open telecommunication standard of cellular wireless technologies, 5G, is estimated to be commercially available around 2020. As in previous cellular telecom standards, 5G is developed within the 3G Partnership Project (3GPP) framework, building on components from 4G/LTE, the prevalent telecom standard. Some of the key characteristics of 5G are higher data rates, shorter delays, and high reliability [39]. Requirements integrated into the new standards reflect the needs of tomorrow, i.e., “a fully mobile and connected society” [40]. Ongoing standardization efforts have defined several scenarios and connected them with sets of requirements [1]. These requirements need several solutions, most of which already partially exist [41].

3. Method and Project Description

The methodology applied to this work is the design science approach. Design science research centers around artificial artifacts, designed and developed with the researchers deeply involved. This methodology has been successfully used in information systems research, in which an artifact can be software, hardware, a method, a theory, etc. [42]. The crucial aspects of design science for information systems research are illustrated in the information systems research framework developed

by Hevner, et al. [43]. An artifact is designed and evaluated in a design/test cycle. The design/test cycle gets its relevance in an iterative process of adapting to real-life business needs and evaluating the results in the intended environment, which, in this case, is the shop floor. The design/test cycle is built on theories drawn from the common knowledge base, and new learnings from the design/test cycle can often result in new theories that should be communicated to contribute to new knowledge and expand the knowledge base. This is called the rigor cycle.

As mentioned above, the 5GEM project investigates how the next generation cellular networks, 5G, can support manufacturing industries in implementing smart manufacturing systems. Part of the investigations was to deploy a cellular network on the shop floor, connect a grinding machine, collect data from the grinding process, and to utilize that data in real-time and analytics applications. The results from the applications provide the system feedback loop that can interconnect with an automated system, e.g., controlling the machine directly, or be displayed to manufacturing operators through a mobile support system (mobile application) already in operation on the shop floor. Beyond the fast and scalable connectivity that the infrastructure provided, this required software and hardware designed to collect and transfer the data between source and utilization, which are the relevant artifacts for this research. To connect a producing grinding machine to a cloud network seems like a straightforward task. However, there are many things to consider regarding the technical aspects. First, there is a question of what data should be collected. Then there is the problem of how to collect that data, and then that data needs to be transferred. Solutions for the latter are the relevant artifacts further described in the results chapter. Figure 2 shows how these artifacts relate to the project goal and infrastructure. The artifacts connect a grinding machine to a mobile operator support system and an analytics platform. Furthermore, they are enabled by the connectivity infrastructure, i.e., the cellular and back-end network, handling the low-level data transmissions.

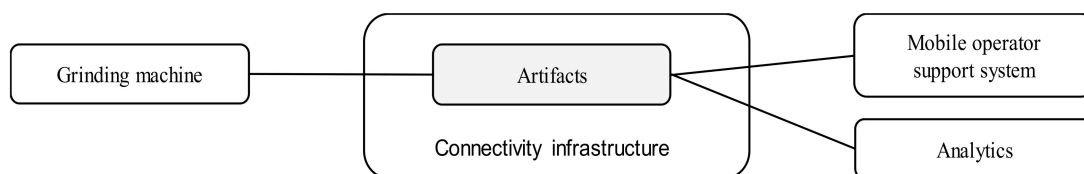


Figure 2. The artifacts described in the results section are the enabling technologies connecting the grinding machine to the data consumers on top of the basic connectivity infrastructure.

3.1. Connectivity Infrastructure

The project infrastructure is distributed over three Swedish cities: Gothenburg, Lund, and Stockholm. See Figure 3 for an overview of the setup. In Gothenburg, the SKF factory is connected to Ericsson's facilities through a radio-link. The Gothenburg site is connected to Ericsson's core network and linked to facilities in Lund, which is a data center that hosts a private cloud in terms of processing and storage. The gathered data is also distributed to the data analytics site in Stockholm. On the shop floor, the closest thing to a 5G network was deployed, which is a dedicated LTE network, TDD (Time Division Duplex) on Band 40, with 5G enabling technologies. Devices are connected to the network through either embedded modem (e.g., LTE enabled mobile phones) or using external USB modems, and the received data rates were measured to 70 Mbps downlink and 17 Mbps uplink.

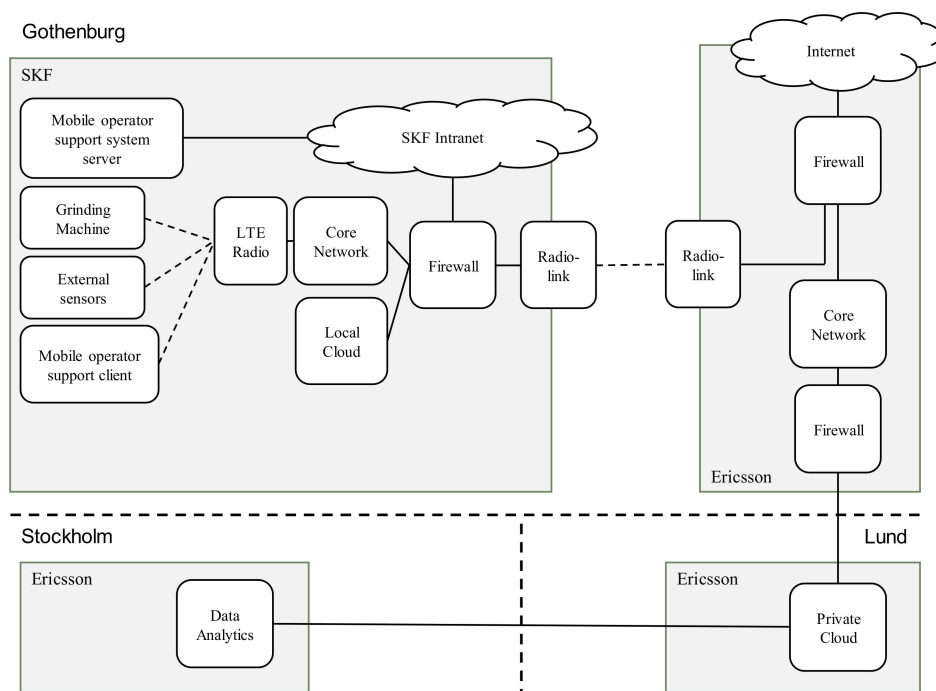


Figure 3. Summary of the project's connectivity infrastructure.

3.2. Mobile Operator Support System

SKF has developed an operator support system with an iOS client [44]. This system can take advantage of a 5G connection in two ways. The first way is to directly connect iOS hardware to a 5G system, which enables the user to send and receive data faster. This is useful when, for example, streaming video instructions. The second way is to interconnect the operator support system with the real-time data collected from the grinding machine and the possible results that come from the data analysis.

4. Results

Data from the grinding process is collected through four separate systems: The grinding machine onboard computer, a vibration measuring system, IO-Link sensors, and an embedded sensor. While it is possible, to some extent, to connect the sensors to the machine PLC (Programmable Logic Controller) and transfer all the data from the onboard computer, that solution was not opted for because it would require extensive change in the machine, a task requiring an expert with very limited time. Each system supports different communication standards and/or methods, which made the overall system more complicated. To simplify this, the actor-based IoT middleware Calvin was used for aligning the data transfer. Calvin is a distributed software system, which allows its function blocks to run on different hardware while communicating seamlessly with each other. The reference implementation of the Calvin platform, Calvin-base [23], runs on most Linux systems, which means that Calvin can run in hardware in the local cloud as well as in the data center.

The different connected systems are summarized in Table 1. The grinding machine computer and the vibration system are connected through the OPC UA standard, which they support to various degrees. The external sensors support the IO-Link protocol, which allows data extraction directly over TCP/IP (Transport Control Protocol/Internet Protocol). However, to simplify the connection to the Calvin actor, another gateway was developed that translates the TCP stream and shares the data as a RESTful web service. This translation software was developed using the Play framework [45], and the raw data could be mapped using the IO-link data sheets. The embedded sensor is a temperature sensor

directly connected to a Raspberry Pi computer. Since Calvin can run on Raspbian, the Linux distribution supported by the Raspberry Pi, a Calvin actor can communicate directly with the embedded sensor.

Table 1. The different connected systems from which the data is collected.

Data Source	Description	Protocol
Machine onboard computer	The machine computer can provide most of the important data related to the grinding process.	OPC UA
Vibration system	The external vibration system consists of vibration sensors that are mounted on the machine, it sends an aggregated version of the vibration data.	OPC UA
IO-Link sensors	IO-Link sensors communicate (using IO-Link) with a gateway, called IO-Link master, from which it is possible to retrieve sensor data over TCP/IP. A second gateway translates the data and makes it available as a web service.	REST API
Embedded Sensor	Temperature sensor connected to a Raspberry Pi computer.	N/A

To connect the systems to Calvin, support for each communication or device was needed. This was achieved by implementing a plugin which handles all communication with the device or service in the platform abstraction layer, called calvinsys. The Calvin application, visualized in Figure 4, is running in a Calvin system spanning three instances: One on a Raspberry Pi located near the machine, one in the local cloud in the factory, and one running in the Ericsson Research data center. Letting the platform handle all specifics of the external service gives the illusion of the service being part of the platform itself. The function blocks, or actors, are distributed on the three calvinsys instances and the communication between them is seamless, meaning that the physical separation of the calvinsys instances is invisible to the function blocks.

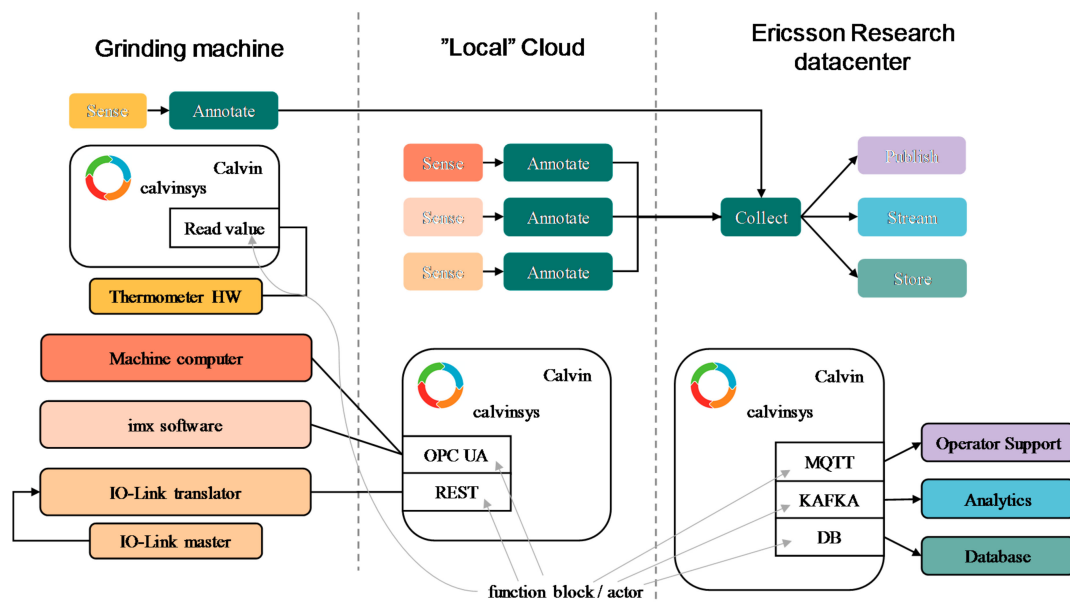


Figure 4. The Calvin application is distributed over three physical locations. One in the Raspberry Pi located by the machine that reads the thermometer values, one in the local cloud in the factory reading data from the machine and external sensors, and finally in the central private cloud to distribute and store the data.

When the data has been transferred to the cloud through the Calvin application, it is distributed to three different data consumers: The database, the analytics application, and the mobile operator support system. The operator support system already utilizes the publish/subscribe protocol MQTT

as the default communication protocol. After deciding on the structure for the topics, a Calvin function block was implemented that publishes the requested datatypes. The data is also streamed to the data analytics center with the help of Kafka [29], which can also fetch stored information from the database if needed.

The database is a MongoDB setup with three replicas, a general-purpose NoSQL database. The data is stored as JSON (JavaScript Object Notation) documents with value, timestamp, and origin of the data as key components. An example entry can be seen in Figure 5. The application collects and transmits 97 different data values, most of them from the onboard computer. With a scan rate of 100 milliseconds, these values generate 16.5 updates per second, which translates to approximately 128 kbps of transmitted data including overhead. The data analytics experts can use the circa 1.4 million daily entries to test and evaluate new algorithms and approaches. At the shop floor, the operators can see and act on a few of these data entries, and remote monitoring and control can now easily be expanded upon.

```
{
  "status": "0, Good, the operation completed successfully.",
  "info": "Position Loading wagon, mm",
  "calvints": 1499990404234,
  "sourcets": "2017-07-14 00:00:04.064000",
  "value": "-10.0036686659",
  "id": "ns=2;s=/Nck/MachineAxis/aaIm [3]",
  "tag": "aaIm_Z",
  "serverts": "2017-07-14 00:00:04.124000",
  "type": "Double",
  "origin": "GU-1"
}
```

Figure 5. Example data entry in JSON format.

5. Discussion

The described artifacts have been implemented and tested in a real-life production system at SKF and have been proven to enable several steps towards CPPS. When 5G networks become available in a few years they will enable fast and scalable connectivity for the shop floor. The question for the 5G-Enabled Manufacturing project was therefore how that connectivity can be utilized. The dedicated LTE network that was deployed during the project was more than capable of handling the presented solution (128 kbps < 17 Mbps), which means that it fulfills the requirement of fast and scalable connectivity. Data is collected from several different systems mounted on the grinding machine, which increase the generalizability of the described solution.

By exploiting Calvin as a common connectivity platform together with open industry standards for data communication and storage, relevant information is made visible in a cloud environment. Kafka and MongoDB are used to stream aggregated data to the analytics center that can apply algorithms and increase system transparency. Since Calvin also aligns the different types of connected systems, both locally in the factory and centrally in the private cloud, it would also simplify the process of connecting more machines and equipment, a prerequisite for more predictive capacity. Furthermore, the mobile operator support system allows manufacturing operators to access real-time data and to share important information.

There are many ways to collect data from machines, and some of the described enabling technologies and their application have been tested before [46,47]. However, when practically collecting data from a machine on the shop floor there seems to be no obvious solution that fits a generic machine setup. Machine suppliers are adding their own support for data collection, but this gives rise to the

issue of data ownership and supporting a multitude of different monitoring systems. Manufacturing companies that collect their own data can be more certain of their own access to it and how it is used. However, some data should perhaps also be centrally collected by equipment suppliers, given the benefit of volume. The solution described here is one generic way to acquire data from machines or equipment for storage in a private cloud and for further distribution.

5.1. System Layout

One important aspect regarding system layout is choosing between the different approaches when it is possible to get the data in different ways. The machine onboard computer is the most important system since it is the source of several datatypes that cannot be collected otherwise. Getting this data proved to be a challenge because the computer had to be reprogrammed to gain access to the desired data. This programming task either directly or indirectly interfered with production since both the machine and the engineer are valuable resources needed to run the manufacturing process. If choosing, e.g., IO-Link instead, the solution is logically decoupled from the physical process.

Calvin is used for aligning all connections to the grinding machine. The result is identical data collection applications over different connections and with high reusability. Another advantage is the ability to run the function blocks, or actors, distributed in the local cloud. Having the platform hosted inside the factory allows the system to function in isolation for a period, meaning it can seamlessly store data to be transmitted later if the connectivity to the data center suddenly disappears [19]. The Calvin application is created by defining the data flows, which is also the approach in other IoT middleware. Creating these data flows can be easier for people with less experience in generic programming languages, which should also be considered when choosing a solution [19]. When scaling the application, platforms like Calvin can also aid in the decision of software deployment using the requirement-based approach [48]. This means that the function blocks can only run where the correct requirement is met, e.g., that there is a specific sensor present.

5.2. Communication Standards

One function block that needed to be developed in this situation was support for OPC UA. While it was straightforward to use existing open source solutions for the actual communication, there were some issues with slight variations in the implementations that had to be considered. Upgrading software in an industrial setting is not always an option as the machine in question is crucial to the manufacturing process. This means that the adaptability of the IoT platform is important.

Two systems support OPC UA as the communication protocol, the grinding machine onboard computer and the IMX (vibration system), which is a natural choice since it is an open, platform-independent system that supports SOA and can enable other future synergies [17,49]. A limitation of OPC UA that was evident during this project is the difference in the various implementations, due to their being self-certified.

With the IO-link enabled sensors it is possible to extract the data directly bypassing the machine computer. That avoids meddling with the machine program, but it can also miss out on the possibility of semantic alignment that adopting the OPC UA standard can enable. In this case, a separate translation software was developed which required some work, but modern software frameworks simplified this task [45]. The solution presented here is a stand-alone REST API supporting the few sensors that were mounted on the machine. Such implementation could be made to support almost all IO-link sensors by connecting it to the common device library service that, at 2017-07-24, had more than 80% of all devices documented [19,50]. Since the presented application was developed, the company ifm electronic GmbH (Essen, Germany) have released gateways with an IoT connection (see Figure 6), which basically works similarly to the REST API that was developed for this application, with a web service exposing the sensor data as a generic JSON (JavaScript Object Notation) document.



Figure 6. On the left: IO-Link gateway used for the described application (ifm AY1020). On the right: Internet of Things (IoT)-enabled IO-Link gateway (ifm AL1930).

MQTT is used to send real-time data to the mobile operator support system. The data is sent using the JSON structure, which is the de facto standard for Web services and aligns well with the document database solution. There is little to say about this, other than that it shows the benefit of utilizing well-defined open standards.

5.3. Open Systems

Without a doubt, the open-source solutions were crucial for the implementations in this project. This is true regarding both the data management infrastructure and end-point solutions. The open systems for process distribution, data streaming, and data storage are industry standard, developed and well tested through the emergence of large and well-known social platforms [51]. For systems like Calvin, the open-source approach is important since existing and future implementations benefit from a large community that can share content. Using Raspberry Pi's or other low-cost microcomputers with attached hardware modules are excellent tools for testing and verifying IoT solutions. However, to take advantage of these systems requires one's own responsibility and overall system knowledge. Traditional industrial systems are hierarchical, centralized, and built for robustness. This leads to stand-alone systems since each subsystem is built fully deterministically and with a well-defined scope. Decentralized systems with more horizontal connections are more flexible and complex [7].

6. Conclusions

A generic solution to collect and send data from a grinding machine, implemented in a real-life industrial setting, has been described and discussed regarding technical interoperability, focusing on the system layout, communication standards, and open systems. The findings are tied to the key question of improving manufacturing performance by connecting the technical implementation to the Industry 4.0 maturity Index model. From the discussion, it can be derived that manufacturing enterprises need to internalize the knowledge of how IT systems can be utilized to connect their existing and new equipment, while not becoming overly reliant on technology providers. The example of IoT enabled IO-Link sensors also shows the fast development of available equipment, but also that this development will most likely align with existing efforts if current standards and state of the art solutions are followed.

Author Contributions: Conceptualization, M.Å.; Writing–Original Draft, M.Å.; Writing–Review & Editing, M.Å., J.S., U.E., D.M., and T.H.; Software, M.Å., O.A., and T.H.; Supervision, Å.F.-B.; Project Administration, J.S., U.E., M.F., M.Å., M.B., and C.L.; Funding Acquisition, J.S., U.E., and M.F.

Funding: This research was funded by the national Digital Industrial Pilot Programme operated by the Swedish Agency for Innovation Systems (VINNOVA) grant number [2015-06755].

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

- 3GPP TS 22.261: Service Requirements for the 5G System; Stage 1; Release 15 ed. Available online: http://www.3gpp.org/ftp/Specs/archive/22_series/22.261/22261-f40.zip (accessed on 20 June 2018).
- Vernadat, F.B. Technical, semantic and organizational issues of enterprise interoperability and networking. *Ann. Rev. Control* **2010**, *34*, 139–144. [[CrossRef](#)]
- Mell, P.; Grance, T. *The Nist Definition of Cloud Computing—Recommendations of the National Institute of Standards and Technology*; 1047-6210; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2011; 7p.
- Chen, M.; Mao, S.; Liu, Y. Big data: A survey. *Mob. Netw. Appl.* **2014**, *19*, 171–209. [[CrossRef](#)]
- Atzori, L.; Iera, A.; Morabito, G. The internet of things: A survey. *Comput. Netw.* **2010**, *54*, 2787–2805. [[CrossRef](#)]
- Schroth, C.; Janner, T. Web 2.0 and soa: Converging concepts enabling the internet of services. *IT Prof.* **2007**, *9*, 36–41. [[CrossRef](#)]
- Monostori, L.; Kádár, B.; Bauernhansl, T.; Kondoh, S.; Kumara, S.; Reinhart, G.; Sauer, O.; Schuh, G.; Sihm, W.; Ueda, K. Cyber-physical systems in manufacturing. *CIRP Ann. Manuf. Technol.* **2016**, *65*, 621–641. [[CrossRef](#)]
- Hermann, M.; Pentek, T.; Otto, B. Design principles for industrie 4.0 scenarios. In Proceedings of the 49th Hawaii International Conference on System Sciences (HICSS), Koloa, HI, USA, 5–8 January 2016.
- Mittal, S.; Khan, M.A.; Romero, D.; Wuest, T. Smart manufacturing: Characteristics, technologies and enabling factors. *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.* **2017**. [[CrossRef](#)]
- Schuh, G.; Anderl, R.; Gausemeier, J.; Hompe, M.T.; Wahlster, W. *Industrie 4.0 Maturity Index: Managing the Digital Transformation of Companies*; acatech STUDY: Munich, Germany, 2017.
- Chen, D.; Daclin, N. Framework for enterprise interoperability. In Proceedings of the IFAC Workshop EI2N, Bordeaux, France, 22–24 March 2006; pp. 77–88.
- Zhang, Q.; Cheng, L.; Boutaba, R. Cloud computing: State-of-the-art and research challenges. *J. Internet Serv. Appl.* **2010**, *1*, 7–18. [[CrossRef](#)]
- Gantz, J.; Reinsel, D. *Extracting Value from Chaos*; IDC Go-to-Market Services: Framingham, MA, USA, 2011.
- Papakostas, N.; O’Connor, J.; Byrne, G. Internet of things technologies in manufacturing: Application areas, challenges and outlook. In Proceedings of the 2016 International Conference on Information Society (i-Society), Dublin, Ireland, 10–13 October 2016; pp. 126–131.
- Gubbi, J.; Buyya, R.; Marusic, S.; Palaniswami, M. Internet of things (iot): A vision, architectural elements, and future directions. *Future Gener. Comput. Syst.* **2013**, *29*, 1645–1660. [[CrossRef](#)]
- Thoben, K.-D.; Wiesner, S.; Wuest, T. “Industrie 4.0” and smart manufacturing—A review of research issues and application examples. *Int. J. Autom. Technol.* **2017**, *11*, 4–16. [[CrossRef](#)]
- Henßen, R.; Schleipen, M. Interoperability between opc ua and automationml. *Procedia CIRP* **2014**, *25*, 297–304. [[CrossRef](#)]
- Wang, L.; Haghghi, A. Combined strength of holons, agents and function blocks in cyber-physical systems. *J. Manuf. Syst.* **2016**, *40*, 25–34. [[CrossRef](#)]
- Ngu, A.H.; Gutierrez, M.; Metsis, V.; Nepal, S.; Sheng, Q.Z. Iot middleware: A survey on issues and enabling technologies. *IEEE Internet Things J.* **2017**, *4*, 1–20. [[CrossRef](#)]
- Derhamy, H.; Eliasson, J.; Delsing, J.; Priller, P. A survey of commercial frameworks for the internet of things. In Proceedings of the 20th Conference on Emerging Technologies & Factory Automation (ETFA), Luxembourg, 8–11 September 2015.
- Ptolemy, II. Available online: <https://ptolemy.eecs.berkeley.edu/ptolemyII/> (accessed on 22 November 2017).
- IBM Emerging Technologies. Node-Red. Available online: <http://nodered.org/> (accessed on 20 June 2018).

23. Calvin—Lets Things Talk to Things. Available online: <https://www.github.com/EricssonResearch/calvin-base> (accessed on 15 December 2017).
24. Latronico, E.; Lee, E.A.; Lohstroh, M.; Shaver, C.; Wasicek, A.; Weber, M. A vision of swarmlets. *IEEE Internet Comput.* **2015**, *19*, 20–28. [[CrossRef](#)]
25. Persson, P.; Angelsmark, O. Calvin—Merging cloud and iot. *Procedia Comput. Sci.* **2015**, *52*, 210–217. [[CrossRef](#)]
26. Yannuzzi, M.; Milito, R.; Serral-Gracià, R.; Montero, D.; Nemirovsky, M. Key ingredients in an iot recipe: Fog computing, cloud computing, and more fog computing. In Proceedings of the 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Athens, Greece, 1–3 December 2014; pp. 325–329.
27. Apache Hadoop. Available online: <http://hadoop.apache.org/> (accessed on 5 November 2017).
28. Apache Spark: Lightning-Fast Cluster Computing. Available online: <http://spark.apache.org/> (accessed on 18 December 2017).
29. Apache Kafka: A distributed Streaming Platform. Available online: <https://kafka.apache.org/> (accessed on 29 November 2017).
30. Nosql. Available online: <http://nosql-database.org/> (accessed on 18 December 2017).
31. Xu, L.; He, W.; Li, S. Internet of things in industries: A survey. *IEEE Trans. Ind. Inform.* **2014**, *10*, 2233–2243. [[CrossRef](#)]
32. Botta, A.; Donato, W.d.; Persico, V.; Pescapé, A. Integration of cloud computing and internet of things: A survey. *Future Gener. Comput. Syst.* **2015**, *56*, 684–700. [[CrossRef](#)]
33. Liu, X.F.; Shahriar, M.R.; Al Sunny, S.N.; Leu, M.C.; Hu, L. Cyber-physical manufacturing cloud: Architecture, virtualization, communication, and testbed. *J. Manuf. Syst.* **2017**, *43*, 352–364. [[CrossRef](#)]
34. Endrei, M. *Patterns: Service-Oriented Architecture and Web Services*; IBM Corp., International Technical Support: Poughkeepsie, NY, USA, 2004; Volume 1.
35. Perrey, R.; Lycett, M. Service-oriented architecture. In Proceedings of the 2003 Symposium on Applications and the Internet Workshops, Orlando, FL, USA, USA, 27–31 January 2003.
36. Fielding, R.T. *Architectural Styles and the Design of Network-Based Software Architectures*; University of California: Irvine, CA, USA, 2000.
37. IEC/ISO 20922. Information Technology—Message Queuing Telemetry Transport (MQTT) v3.1.1. Available online: http://www.iso.org/iso/catalogue_detail.htm?csnumber=69466 (accessed on 20 June 2018).
38. Shelby, Z.; Hartke, K.; Bormann, C. *The Constrained Application Protocol (CoAP)*; Internet Engineering Task Force: Fremont, CA, USA, 2014.
39. 5G Radio Access—Capabilities and Technologies. Uen 284 23-3204. Available online: <https://www.ericsson.com/en/white-papers/5g-radio-access--capabilities-and-technologies> (accessed on 21 June 2018).
40. Ngmn 5G Whitepaper. Available online: https://www.ngmn.org/fileadmin/ngmn/content/images/news/ngmn_news/NGMN_5G_White_Paper_V1_0.pdf (accessed on 21 June 2018).
41. Akyildiz, I.F.; Nie, S.; Lin, S.-C.; Chandrasekaran, M. 5G roadmap: 10 key enabling technologies. *Comput. Netw.* **2016**, *106*, 17–48. [[CrossRef](#)]
42. Hevner, A.R.; Chatterjee, S. *Design Research in Information Systems: Theory and Practice*; Springer: London, UK; New York, NY, USA, 2010; Volume 22.
43. Hevner, A.R.; March, S.T.; Park, J.; Ram, S. Design science in information systems research. *MIS Q.* **2004**, *28*, 75–105. [[CrossRef](#)]
44. Åkerman, M.; Fast-Berglund, Å.; Karlsson, M.; Stahre, J. Introducing customized ict for operators in manufacturing. *Procedia CIRP* **2016**, *41*, 490–495. [[CrossRef](#)]
45. Play. Available online: <https://www.playframework.com/> (accessed on 28 February 2018).
46. Schlechtendahl, J.; Keinert, M.; Kretschmer, F.; Lechler, A.; Verl, A. Making existing production systems industry 4.0-ready. *Prod. Eng.* **2014**, *9*, 143–148. [[CrossRef](#)]
47. Mourtzis, D.; Vlachou, E.; Milas, N. Industrial big data as a result of iot adoption in manufacturing. *Procedia CIRP* **2016**, *55*, 290–295. [[CrossRef](#)]
48. Angelsmark, O.; Persson, P. Requirement-based deployment of applications in calvin. In Proceedings of the Interoperability and Open-Source Solutions for the Internet of Things: Second International Workshop; Springer: Cham, Switzerland, 2017; 10218, pp. 72–87.
49. Unified Architecture. Available online: <https://opcfoundation.org/about/opc-technologies/opc-ua/> (accessed on 21 November 2017).

50. What Is Io-Link? Available online: http://www.io-link.com/en/Technology/what_is_IO-Link.php?thisID=76 (accessed on 30 August 2017).
51. Sumbaly, R.; Kreps, J.; Shah, S. The big data ecosystem at linkedin. In Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data, New York, NY, USA, 22–27 June 2013; pp. 1125–1134.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).