OPeRAte: An IoT Approach towards Collaborative, Manufacturer-independent Farming 4.0

Maik Fruhner, Thorben Iggena, Franz Kraatz, Frank Nordemann, Heiko Tapken and Ralf Tönjes
Faculty of Engineering and Computer Science, Osnabrück University of Applied Siences,
Albrechtstr. 30, Osnabrück, Germany

Keywords: Agriculture, IoT, Precision Farming, Farming 4.0, Big Data.

Abstract: Without modern agricultural technology, it would not be possible to feed the world's steadily growing population. In order to be able to handle this task in the future, new improvements must be constantly developed to improve agriculture and increase yields. These include methods known as 'Precision Farming', 'Smart Farming' or 'Farming 4.0'. These terms describe working in the field with high-tech machines that are supported by intelligent systems that communicate with each other. But at the present time, such intercommunicating ecosystems are manufacturer-bound and hardly interoperable. This paper presents a new approach to connecting various agricultural machines from different manufacturers into a common network of IoT devices. In this project, a framework for the orchestration of agricultural processes is being developed that is capable of planning, controlling, monitoring and documenting joint collaborative tasks between many independent machines while breaking the commitment to a single manufacturer. The first application example of the system is the development of a tank trailer for liquid slurry spreading with various sensors and controls. By using IoT-specific technologies, the tank can already be configured by a process management system, so that exact nutrient quantities are applied part-field-specifically and a legally compliant documentation is generated.

1 INTRODUCTION

The world's population has grown permanently in the past and will continue to do so in the coming decades (United Nations, Department of Economic and Social Affairs, Population Division, 2017). This ever-increasing number of the human population simultaneously demands a higher food production and results in a steady decline of arable land per person (DBV, 2018), driven by the necessary expansion of urban areas and infrastructure. The required increasing agricultural yield can only be achieved through the development of special optimization techniques in the agricultural sector.

A common problem in trying to improve yield is the variety of devices required and the associated proprietary technology that leads to cross-manufacturer incompatibility. Approaches that try to solve this problem often use old, outdated technology that is no longer state of the art. The use of modern cloud architectures for agricultural use is at most conceptually designed, but still years away from market maturity.

This paper presents a novel approach towards optimizing complex agricultural processes by connecting individual machines, regardless of their vendor, into a large network of IoT devices. The cross-process control via a management system makes it possible to assign tasks to any number of vehicle/trailer combinations so that orders can then be processed jointly but still autonomously. The aim of the project is the creation of a framework for the management of cross-actor, cooperative processes in agricultural engineering. The combination of several modern technologies results in a continuous cloud based data flow between all machines and a process management system that not only simplifies distributed tasks but also legally secures the responsible persons.

The current project uses slurry spreading as the research example. The new aspects developed in the project are explained in the paper on the basis of this example.

After the following presentation of the motivation and the current state of the art, the most important aspects of the project are presented in the form of the system's architecture and implementation. An evaluation of the current state of development illustrates the progress achieved.

Fruhner, M., Iggena, T., Kraatz, F., Nordemann, F., Tapken, H. and Tönjes, R.
OPeRAte: An IoT Approach towards Collaborative, Manufacturer-independent Farming 4.0.
DOI: 10.5220/0007760101650176
In *Proceedings of the 4th International Conference on Internet of Things, Big Data and Security (IoTBDS 2019)*, pages 165-176
ISBN: 978-989-758-369-8

2 MOTIVATION

A key word to mention nowadays in connection with agriculture certainly is 'Precision Farming'. The term describes the systematic cultivation of farmland, depending on current soil conditions. An important factor in agriculture is the fertilization of the fields. Without precise knowledge of the soil condition and the nutrients it contains, it is not possible to give accurate information about the quantity of fertiliser to be applied. In addition, the soil of a field is not the same in every location. In fact, a field usually consists of many different soil types and nutritional values. The values change constantly between the cultivation periods, depending on the currently sown crop and the crops of previous years.

This is where the OPeRAte project comes into place. It supports traditional agricultural practices, such as slurry spreading or maize harvesting, with high-tech, automated processes and machines operating as members of the Internet of Things.

In concrete terms, it represents a framework for orchestrating agricultural processes at three successive layers:

- The **machine layer**, executing commands on a single machine,
- the **task-bound collaboration layer**, controlling multiple machines and devices executing a common task
- and the **joined collaboration layer**, controlling many machines and devices across different tasks.

The framework is aimed at farmers and contractors who wish to run an automated, resource-efficient, part-field-specific farm management system across all three mentioned layers. This means, for example, to automatically calculate the correct, varying quantities of fertiliser to apply to different parts of a field and sending this information to the corresponding machine to execute. Fertilizing part-field-specific results in higher quality and quantity yield.

In order to put this intention into practice, the agricultural machinery must become part of a network in the sense of the Internet of Things. Due to the continuous communication between the devices, each machine plays its own role in the system. Part of the commnication is the exchange of status and sensor data between machines and the management platform.

This, in turn, leads to the generation and maintenance of large amounts of data, which can later be used for evaluation and consequential ressource optimization. These can be 360-degree views of the data, the fully automated generation of application maps and legally required documentation, or the agile replanning of an ongoing process for an overall better result.

A cultivation cycle ranges from sowing, fertilizing and treatment with pesticides to harvesting. The scenario currently being worked on is a precise, automated fertilization through spreading of slurry. While using the OPeRAte-Framework, the farmer benefits in many ways. The two main issues are monetary and legal aspects and come to bear as follows.

When a farmer cultivates his field, he wants to make the most of his time and resources. Depending on the condition of the soil, fertiliser must be applied to the field in many different quantities at different locations. Specific application maps used for Precision Farming generated by the OPeRAte Process Management describe these output ratios visually for the farmer to understand and technically for the application through the slurry tank. When an intelligent tank reads the output values from an application map, it can automatically regulate the current flow rate of the fertilizer. These maps - while being able to automate the process of fertilizing - also conform to the german fertilizer ordinance which came into force in 2017 (BMEL, 2017).

Therefore, a precise documentation of the applied amounts of slurry and the resulting soil conditions can be generated automatically after each completed task. The user then has the ability to easily prove all relevant aspects to the government.

The first goal of the OPeRAte project is the prototypical development of a high-tech slurry tank, which can be integrated into this highly distributed system and which considers all mentioned aspects.

3 RELATED TOPICS

The goals pursued in this project are not entirely new. International manufacturers of agricultural machinery offer similar functionalities for cross-process communication and interaction with their equipment. However, these solutions are mostly proprietary, so that they can only be used in the manufacturer's own ecosystem of devices. Compatibility with other vendors or even open interfaces are available in the rarest cases.

3.1 ISOBUS

An already widely used standard for communication across agricultural machinery is called ISOBUS (AEF, 2019), which is the name of the ISO 11783 norm (ISO, 2017). It is currently being maintained by the Agricultural Industry Electronics Foundation (AEF).

It not only represents a communication bus, but also serves, for example, to control the task controller or the universal terminal and as a result, it enables the control of several connected devices via a single terminal in the driver's cab.

The ISO XML format defined in the same standard is used as the data structure. It enables cross-device planning, control and evaluation of agricultural processes. This allows new tasks to be created in the farmer's management system and then to be copied to a tractor's terminal. The results can also be transferred back to the management platform.

Although it is present in virtually every modern agricultural machine, its major drawback is that the standard is too loosely defined, so that each manufacturer can implement it in slightly different ways, leading to incompatibilities within the standard.

Developers and manufacturers must license the definition of the standard in order to gain insight into the details and thus be able to program interfaces.

The AEF also defined EFDI (Extended FMIS Data Interface) (Schlingmann, 2016), a standard for wireless communication across ISOBUS devices. It could one day replace the data exchange via USB flash drive, but has not yet become widespread.

3.2 Open Communication

For several years now, among others the Osnabrück University of Applied Sciences, together with various partners, has been researching possible solutions for open and interoperable communication interfaces.

The completed 'KOMOBAR' project (Freye, 2010) focused on the development of a reference application for the self-optimization of supply chains. The optimisation of logistics chains should reduce fuel and personnel costs as well as pollutant emissions, increase energy efficiency and improve the overall quality of the products concerned. These objectives were to be achieved through the development of open APIs, which would subsequently be available to SMEs. However, the focus was solely on logistics. Optimizations for the farming process itself were not considered.

The ongoing project 'Smart Data, Smart Services' (DFKI, 2017) is in the process of developing an 'agricultural data hub'. The goal is to define and implement an open ecosystem that enables data to be exchanged and evaluated independently of manufacturers and services in various different data formats. A support for controlling complex processes, as proposed in this paper's project, is not part of the research, though.

Both research projects attach great importance to the fact that the interfaces are and remain freely accessible and are therefore intended to overcome the constraints imposed by the manufacturers of proprietary systems.

3.3 IoT in Farming

When it comes to the use of IoT in agriculture, systems for stationary sensors are found almost exclusively. Many companies offer their own commercial solutions for monitoring a variety of different kinds of data such as weather and soil conditions.

More advanced use of IoT in farming have been demonstrated, for example, in the form of a small connected smart farming system (Minwoo Ryu et al., 2015) or a feed silo measuring the quality and quantity of its content (Agrawal et al., 2016).

However, large-scale IoT frameworks capable of managing complete farms have only been researched for a few years and have not yet entered the market (Kamilaris et al., 2016). The authors therefore developed their own framework, which is able to process sensory data of up to 300 sensors deployed at a field and perform reasoning in real-time on the data. The farmer then receives recommendations for decision-making during the execution of a job.

When it comes to IoT devices on agricultural vehicles, researchers still have to develop new ideas. As stated above though, manufacturer-bound commercial solutions are already on the way. Current vehicles produced by John Deere already come with connectivity systems such as LTE, Wifi and Bluetooth and can communicate with their own cloud (Ferguson, 2018) and with each other.

The project proposed in this paper goes one step further and enables this communication in an open, transparent way. The aim is to enable joint communication and work between machines, even with equipment from different manufacturers, and thus to eliminate the need for a single ecosystem.

4 ARCHITECTURE

The following chapter describes the fundamental architecture of the OPeRAte-Framework. As stated above, the system's process management acts on three superimposed layers, which are shown in Figure 1. In the subsequent sections, the indiviual features of the three distinct layers are explained in detail from bottom up.

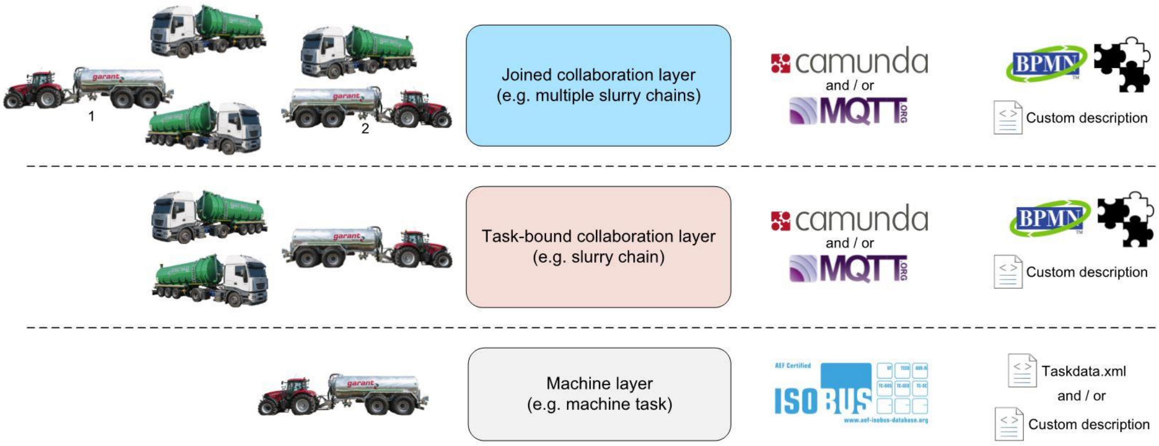

Figure 1: The three layers of the OPeRAte Process Management at the example of slurry spreading.

4.1 Machine Layer

The most basic processing layer of the system is the machine layer. It lays the foundation of the layered system by representing the bottom level of task execution by describing the capabilities modern agricultural machines already possess.

In particular, it describes and covers the functionality of a single tractor-implement combination. At this level, for example, the previously mentioned application maps or task data are generated and used.

In order to make the tasks machine-readable, the process management must output the information required for a job in a certain structure. This is accomplished by using the established 'Taskdata.XML' format defined in the ISO 11783 norm (ISO, 2017), which every Electronic Control Unit (ECU) can understand.

The Taskdata file format is used to define automation processes for field cultivation. For example, a task can describe what kind of work has to be done on a particular part-field. The field boundaries, the job to be done and the necessary parameters are included so that a job can be automated without further configuration.

Given a set of input parameters, the developed process management system is capable of automatically creating such a standard-compliant XML file, which defines an entire task execution.

A task is assigned to one or more vehicles which are in charge of fulfilling this job. The definition can then be sent to the assigned machines, where it can either wait for the driver to initiate it, or it can be executed autonomously immediately or at a later point in time if the vehicle is capable of self-driving.

At this physical level, all communication takes places via a standardized ISOBUS connection. All modern agricultural machinery support the ISO standard mentioned above, which enables unambiguous control of vehicles and equipment.

The machine layer takes care of the definition and control of a task for a single vehicle combination.

4.2 Task-bound Collaboration Layer

During the execution of a task though, vast amounts of data are produced by the variety of sensors connected to the tractor and the implements. By default, this data is recorded and saved on device for the farmer to export via USB after a task is complete.

The task-bound layer now extends this approach so that the data is captured not only offline, but also online. This means that all sensory data can be sent to the process management system in real time during acquisition.

For this purpose, several new exchange formats have been developed. One of the main objectives of the project is the establishment of manufacturer-independent interoperability. This can only be achieved by converting the jobs created by the process management into the different proprietary file formats. An internal description language handles the conversion of sensor, status, log and task data.

This data transmission is performed under the aspect of a publisher-subscriber schema using the MQTT protocol, which will be described in detail in section 5.3. In principle, the data is published by an IoT device at a certain 'topic' and enables interested, authorized parties to subscribe to this continuous stream of new information.

One of those subscribers is the process management. It collects and aggregates all received data in order to perform various different types of processing on them. One example would be the generation of a complete and comprehensive documentation of the work done once a task has been completed.

However, it is not unusual in agriculture for several vehicles to be involved in the same task. This is also known as a task chain. Staying in the slurry spreading scenario, there might be one party that delivers the fertilizer while the second party actually applies it to the field. In order to realize the possibility of carrying out this task-bound collaboration, detailed communication is required not only between the process management and the vehices, but also between the vehicles themselves.

The requirement to combine different subtasks into a single super-task while at the same time allowing the indiviual machines to work autonomously can be regarded as a rather complex challenge. To overcome this, the task chains are modelled as deterministic processes. The Business Process Model and Notation (BPMN) is used to formally describe them while remaining hidden to the end user. A farmer, for example, never has to interact directly with BPMN, but instead uses a convenient, easy-to-use web front-end.

There, the client is able to configure an upcoming task by defining a set of parameters, such as:

- the field to fertilize
- the tractor and implement to use
- the location to acquire the fertilizer from
- last year's yield

After completion, the process management can automatically send the newly created task data to the corresponding devices. In addition, a new MQTT topic will be created, which each participating device should subscribe to in order to get all relevant information about the current task.

Through this digital networking where sensor and status data is exchanged at all times, the process can be monitored and controlled in real time. This allows a dynamic adaptation of the running processes, which might be necessary due to disfunctions of a device or manual reconfiguration through the responsible person. Again, updates are sent out immediately to ensure optimum yield.

Figure 2 shows the graphical visualization of the monitoring. A tractor symbol indicates the current position of the machine. The red polygon illustrates the field border of the current task. The green line highlights the chosen route.

Such an integration of MQTT and BPMN into the field of agriculture is completely new. Through formalized data structures, it is possible to connect many manufacturer-independent devices IoT-controlled to a large, jointly working network.

Figure 2: A process can be monitored on a live map.

4.3 Joined Collaboration Layer

The final and most advanced layer deals with so-called joined collaboration. At larger farms, work may not only be done on one collaborative task, but on many concurrent tasks. These simultaneous tasks, although not directly interdependent, can also interact with each other thanks to OPeRAte.

Interprocess communication can be useful if, for example, a process chain reports an imminent delay caused by an unforeseen event. In this case, another operational process chain can delegate a currently dispensable machine to the chain fallen behind in schedule to make up for the loss.

Together, these three layers form a unit with which the process management system can configure, control and monitor several active process chains across different machines and fields.

5 IMPLEMENTATION

After introducing the basic idea and highlighting the archtitectural design, the next chapter will give an insight into some of the technical details of the OPeRAte project. Starting with the implementation of the process management, the section also shows how the large amounts of data (considered as Big Data) are stored correctly. It then describes the communication between all IoT devices, servers and other clients, and concludes by explaining how all data transmissions are secured against unauthorized access.

5.1 Process Management

The three controlling layers of the process management form the core of the OPeRAte-Framework. The management system is used to define, configure, initiate, monitor and evaluate tasks (Nordemann et al., 2016).

Always hidden from the end user, it processes initiated tasks on a model basis. This means that each executable process performs as a self-contained task. A process can consist of various subprocesses and can itself be a subprocess of a higher-level process.

As mentioned before, all possible processes are created as BPMN flow diagrams. The advantage of the individual sub-processes is that they have carefully been implemented in such a way as they can be reused to describe another super-process in a completely different context, consisting of a different set of sub-processes.

The Camunda software serves as a modeling framework. This not only makes it possible to plan and define processes, but also to run them directly in the execution environment.

The so-called 'workflow automation' offers an API for the execution of external jobs. A web frontend, for example, can then start an operation through a REST request. The modeled flowchart is processed and returns a response to the frontend. This is noteworthy from a technical point of view, since graphically modeled processes usually only serve to illustrate an implemented algorithm and cannot be executed themselves. Furthermore, there is no known project which used BPMN in an agricultural context before.

All business logic of the system is implemented in this way. From planning, monitoring and dynamic rescheduling of running tasks to automated generation of documentation for the target and actual status, everything is implemented via BPMN.

As an enhancement, processes can also be executed in different variants. Based on the input parameters, a diagram can be configured either automatically , semi-automatically or manually.

The default behaviour on one hand is to manually plan and configure a new task via the web interface. The relevant locations, fields and devices are set and the task is registered for a start at a later point in time.

On the other hand, a farmer can start a job by turning on the tractor without a prior configuration of a task data. The device then reports to the process management that a new task was launched. The system automatically registers this task and starts logging the incoming sensor and log data.

Figure 3 illustrates a shortened version of the process model for slurry spreading. The main process consists of five building blocks. The cogwheels indicate a subprocess being responsible to return a result for said block. In general, there is a common description for each process and what it is supposed to do, for each actor involved. The details on how an actor implements his or her subprocess is irrelevant for the other parties, as long as it delivers the desired result.

The first two subprocesses communicate with external services via REST, one being developed by a project partner. First, the application map for automatic adjustments of the liquid fertilizer flow is calculated. Input values given by the farmer through the web interface are used to initialize the request. It does not matter to the system how exactly the map is being generated. This blackbox approach allows external companies to offer their own services to the project's system while at the same time keeping their coporate secrets save. As long as the defined interfaces are used, different implementations of a subprocess can exist.

After converting the remaining input into an ISOXML format, the task is deployed to the machines via MQTT. The required topics are created and the machines subscribe and start to send status updates.

The process keeps looping as long as the slurry spreading is not done. During the loop, status messages are received and handled.

Once all devices report a finished task, log messages are collected and the documentation is being generated. Again, depending on type of task, used machines and location of the field, the implementaion of the documentation process could be varying in many different ways to output a different version for each possible configuration.

But not a single task could be accomplished by the Process Management itself. The second important component is the data storage. A stable communication between all IoT devices and the cloud is not to be forgotten.

5.2 Data Management

In order to be able to analyse completed tasks, find optimizations for upcoming tasks or to retrieve new external information, any data generated by the process management or by the communicating devices must be stored for later use.

The foundation for the data structure is formed by the taskdata standard defined in the ISO 11783 norm. Based on this XML definition, a relational database schema was developed that is able to map incoming task data files to new or existing database entities in a highly normalized database.

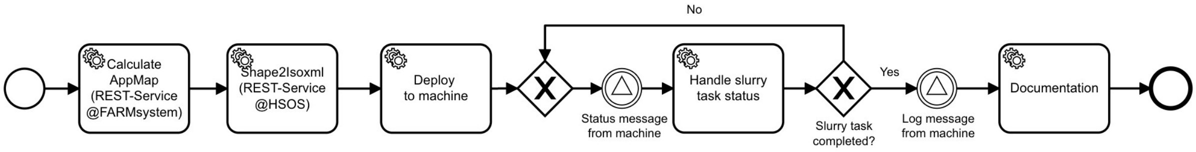

Figure 3: Slurry spreading as an example of a BPMN process model.

5.2.1 Raw Data

Incoming data encoded as Taskdata.XML is considered 'raw' data. This data is read from the XML file to be then written to the raw-database. But this data may occasionally be incorrectly formatted or not conform to the standard.

For this purpose, a parsing software was developed that takes a set of task data as input. The input files are then validated against the standard and corrected if necessary. Finally, the program generates the corresponding entities in the database, resulting in a - potentially adjusted - copy of the information, stored in relational database tables.

Luckily, these do not have to be created by hand, though. The parser written in Java is also able to figure out the database schema and deploy it to the desired database using an XML Schema (XSD) file that describes the structure of the task data input. Using JPA Annotations, all relevant XML-tags are converted to Java entity-classes. These are used in turn to generate the corresponding database tables in PostgreSQL using JDBC.

Saving all task data as new, individual units, however, creates a problem with data integrity. Each task data contains all the partfields, machines and implements among other things used for a job. This data is not kept consistent across multiple task files, though. Each task begins assigning IDs starting with 1, which might result in different identifiers for the same machine in each task, depending on the order they appear in. If now, for example, the same tractor is used twice in two different tasks, it is also stored twice to the database.

This does not correspond to the principles of relational databases and is therefore not a reasonable strategy to keep the data. To solve this problem, the enriched database is introduced.

5.2.2 Data Purgement

In the first transformation step after the import, new data must be revised so that no duplicate entries occur. Recurring data records such as farms, fields, machines and implements are called 'static data at rest' in a second, enriched database. After input into the database, these data records are given a new ID, which differs from the one given by the task, but which will stay unique in the database.

When importing new task data, a mapping must therefore determine whether an entry already exists in the database or not. If so, the record is not saved again, but instead is referenced via foreign key.

After successful reference of new data to existing data, various table columns of the new data coming from the raw database tables are transformed for better further processing. One of these transformations takes numeric columns representing coordinates and stores them in a specific geolocalization format provided by the PostGIS extension for PostgreSQL. This allows fast and lightweight calculations of distances and areas, which is especially advantageous for handling additional data from external sources.

5.2.3 Data Enrichment

At this point, the imported task data was correctly integrated into the existing data records and selected columns were extended to new data types by a systematic combination. In the following data enrichment step, all available information - new and old - will be augmented by adding details from external services.

By using openly available data sources, the generated data for a field can be supplemented in a variety of ways. In the current stage, the imported task data are enhanced by historical weather data provided by the German Weather Service (DWD) and satellite imagery of 13 different spectral bands recorded by the Sentinel 2 program.

One hurdle are the many different data types in which the different information is available. The weather data are provided in grids of thousands of square cells covering the whole of Germany. Hyperspectral satellite data from the Sentinel 2 program are recorded for most of the Earth and are available in different frequency bands and resolutions via various pre-processing APIs.

To make use of these offerings, requests for the latest data are regularly sent to the different services. Certainly not everything that is available is of interest. Only information about the farms and fields present in the system are fetched. If updates are available, they are pulled and matched to the corresponding database entries.

For this procedure, additional database tables are required to save the enrichments. Having analysed the structure of each indiviual external data format, the database is capable of retaining the new information.

After the integration step of external data, static data at rest is not only correlated across different tasks, but also enriched by new information from other sources than the input. With this data storage system, it is possible to merge historical weather data of a certain field with the hyperspectral frequency bands of the corresponding coordinates at a given point in time, for instance to determine the result of a specific fertilisation. This huge amount of correlated data can further be used to run automated analysis of the condition of the farmers' fields and machines.

5.3 Communication

In order to create a dynamic and productive network between all the devices and machines involved in these complex processes, optimal communication is essential. The project makes use of several different communication methods and protocols for different applications: The ISOBUS standard is used for the local communication of a single machine-implement combination. An MQTT Client-Broker architecture offers a reliable data transmission between all mobile and stationary devices and machines involved in a process. RESTful APIs using the common HTTP protocol connect the IoT devices with the process management and the backend.

5.3.1 ISOBUS

Almost every agricultural machine produced in recent years supports the ISO 11783 standard, also known as ISOBUS, managed by the AEF. The ISO standard is a composition of several components. It describes the physical connectors on the machines as well as the file formats and network interfaces used for communication on a machine-implement combination. This means, for example, that the trailer circuits can be controlled via the terminal in the tractor cab.

OPeRAte's goal now is to connect the local communication of the machine layer to a cloud infrastructure using IoT technologies. Addressing the individual, working machines via a central control panel makes it possible to coordinate many individual carriages into a common task. This is done via the MQTT protocol displayed below.

5.3.2 MQTT

The MQTT protocol, based on the Publisher-Subscriber approach, has been a standard in IoT for many years. One reason for this is the smaller overhead and the correspondingly higher performance of data transmission compared to the widely used HTTP protocol (Yokotani and Sasaki, 2016).

The ability to send and receive messages even when network connections are weak becomes very important in agriculture, where machines may not be connected to a cellular network while processing tasks in a field.

All participating devices continuously send sensory and status data to the broker. To ensure that each message reaches all recipients for whom it is relevant, specific topics are defined for each participating device and each task. If a device is in need of information from another device, it can subscribe to one of its topics to get that information, if it has the permission to do so. One of these receivers is the process management. Each device and every task - running and completed - has its own status topic to keep the system informed of its current status.

If, for example, a machine is sending the information that it is currently unused, the process management can assign a new task to it. Therefore, a new topic will be created if not already done and the machine is told to subscribe to it. The machine will read and execute the received task data and send its logs to the new topic created for that particular task.

The transmitted data is structured using the JavaScript Object Notation (JSON). This universally compatible format can be serialized and deserialized on practically any device, making it suitable for system with any computing power.

This procedure on its own is already a significant step forward from the currently established standard. In most systems currently in use, all task data is transferred manually to the agricultural machinery by the farmer via USB flash drive. With the OPeRAte-Framework this procedure becomes obsolete.

5.3.3 REST (HTTP)

Not only the IoT devices have to communicate with each other and with the process management, but also the end user needs a way to interact with the system. Furthermore, the system itself must access the database described above in order to persist and query its data. For this purpose dedicated web services provide REST APIs which offer specific functions for the required operations.

To abstract the management layer - which might appear to complex for an end user - a graphical user interface is offered in form of a dynamic web based frontend which can be used with any modern Internet browser. The web application can be used to manage and configure all registered machines, devices and tasks.

On the backend side, another service is responsible for importing new task data in the form of the standardized ISOBUS XML data format as described in section 5.2.

5.4 Data Security

As shown in section 1 agricultural processes are under a steady technological change, which is on the one hand related to new technologies and possibilities and on the other hand conditional upon changes in law. During modern agricultural processes a huge amount of data is being generated by different actors or more technical roles. As it is unusual that one process can be done by one single role security problems become a major issue as things like authentication or authorization have to be considered during design of new programs or frameworks.

The agricultural framework presented in section 4 is designed to allow different roles to be combined in one process. The process management as the central part is therefore responsible to bring different roles together and to act as a distributor for the data needed by the process.

5.4.1 Security Analysis

To build a secure system with a high user acceptance the design principles of Privacy by Design (PbD) shell be considered (ENISA, 2014), (Cavoukian, 2009). Therefore in a first step the proposed architecture has to be analyzed and extended by security mechanisms before realization. The identified main aspects are:

User and Role Management. Processes in the agricultural environment are often distributed over different participants. Even a small process normally consists of a farmer as the responsible owner and manager of the process, an agricultural contractor, and the workers of the contractor. Each of these roles is supposed to have different access rights within the framework and to guarantee its identity to systems and roles it has to communicate with.

Communication. As the components of the framework are loosely coupled there is a lot of communication between different systems. The communication is realized mainly by REST interfaces and the message bus protocol MQTT as described in section 5.3.2. For the security aspect it has to be ensured that no third party can access the communication interfaces.

Resource Requirements for Processes. As stated in section 5.2 the project is a big data project due to the immense amount and variety of data in the agricultural environment expected in the future. While dealing with data it is important to have a concept of security for the data of a user stored in the projects database.

5.4.2 Security Solutions in Framework

Authentication

To identify the participants of the agricultural processes, the framework uses a certificate based solution. By providing all framework components with certificates the clients can ensure that they are communicating with the right party. In addition this also enables the framework to use SSL/TLS based encryption for communication channels. By also equipping the clients with client certificates the framework components themselves can verify that they send process information to the right participants. For the MQTT communication the framework makes use of client authentication by client certificates combined with passwords. Through additional extensions or extended enterprise versions of MQTT brokers (e.g. HiveMQ (HiveMQ, 2019)) it is also possible to authenticate by external data bases or services like LDAP, which might be used in future versions.

Authorization in Processes

The process communication is realized by MQTT. To avoid unjustified access to MQTT topics belonging to a process by third parties a flexible solution similar to ACL has been integrated. Whereas the usage of file based ACLs is not really adoptable for the processes, as authorizations have to be changed for every new process, an authorization plugin for the mosquitto (Eclipse Foundation, 2019) broker has to be used. This addon enables the broker to query data bases or an ldap for user authorizations.

With this extension the solution of job topic switching depicted in Figure 4 can be used. For a new process the process management creates a new job uuid and selects the participants, which are then added to a new ACL allowing them to access the MQTT topic used by the process. The participants itself are informed about the new topic and subscribe to them. Any other roles would not be able to access the job topic.

Communication Security

The process management as the central element of the framework is responsible for the communication with all needed components. As shown in 5.3.2

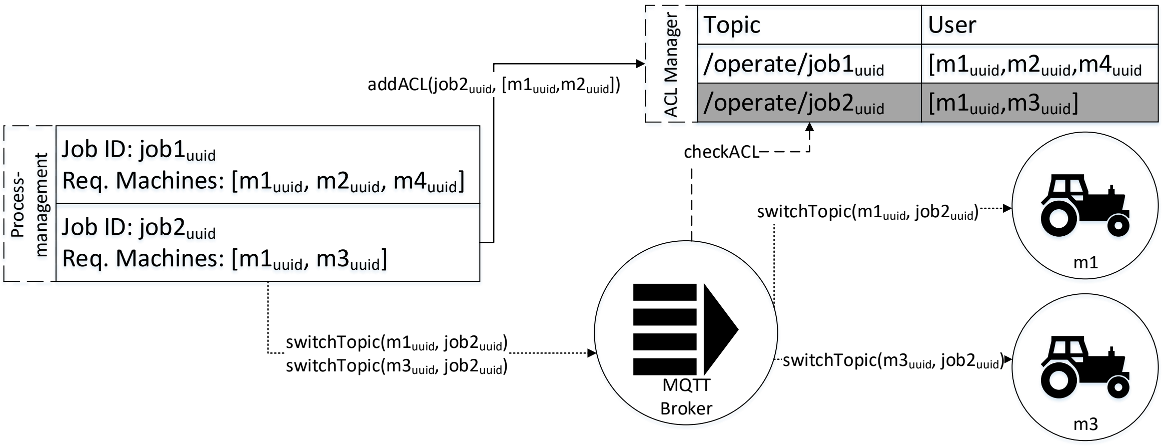

Figure 4: A Visualization of Topic Switching.

the vehicle communication is realized with MQTT. This lightweight protocol for IoT offers options to secure communication connections. To ensure a secure communication between devices and components via MQTT transport encryption will be used. The OASIS MQTT standard allows to use TLS for transport layer. By providing a certificate to the MQTT broker TLS can be enabled (OASIS, 2014), (Dierks and Rescorla, 2008).

Resource Management

As shown in section 5.2 the framework is designed to hold large amounts of data for further optimization. Therefore, the framework has to ensure that only owners of data can decide on how to use them. This requires a strict division of the stored data, which is ensured by policies, also known as row level security. This approach allows to fulfill the design principles of PbD.

6 EVALUATION

The framework presented here has not been just a concept for quite some time now. Through cooperation with some well-known German companies in the agricultural sector, the developed solutions could be implemented in software and hardware. In the current phase an implement is developed, which is supposed to support all features listed in this paper.

More precisely, an intelligent tank trailer for slurry spreading is being engineered. The first version, depicted in Figure 5, is already in prototypical use and tests the functionality of the joined processes on actual fields.

Initial trials are already showing promising results. The device can currently read in an application map and identify the various output values for the different locations in a field contained therein. Then, during the task execution, the values to be applied to the current position are determined with the help of GPS. The tank automatically performs the appropriate hydraulic settings to reduce or increase the flow of the liquid fertilizer. The controlling - as defined by the machine layer - is done via the ISOBUS standard, once the task data was loaded to the tractors main terminal wirelessly through MQTT.

Figure 5: The OPeRAte slurry tank attached to a tractor.

During the task execution all sensory, status and log data are sent back to the framework. The data collected during the jobs can be then used to suggest improvements to the process itself and to the hardware and software. Finally, a legally compliant process documentation is created as a PDF file.

The product has already been presented at many international exhibitions and has always generated great interest and approval. Examples include AgEng 2016 and the International Green Week 2019.

Due to this sucess, the second generation of the tank is already being conceptualized and might soon be produced by the cooperation partners. Once the project has been completed, the manufacturers will use the research results to develop marketable products.

7 CONCLUSION

In this paper, the OPeRAte-project was presented in its current stage and in terms of its long-term goal. The result is a framework for platform-independent, cross-vendor execution of joined, distributed agricultural tasks.

At present, there is a BPMN-based three-layer process management which is responsible for creating, controlling and monitoring task collaboration. As a reference task serves liquid slurry spreading.

Special hardware in the form of a tank trailer was manufactured by cooperation partners and successfully put to the test in some initial runs. The connection of all machines, devices and sensors to the Internet and the contiuous flow of information between them creates a central system for the coordination of multiple task chains. End users, such as farmers and contractors, benefit from dynamic resource optimization and automated generation of legally required documentation of tasks performed. At the same time, complex, internal process structures remain hidden from the user and are instead abstracted in an understandable way by logically structured, graphical user interfaces.

During development, care was taken to keep the software architecture as generic as possible so that modules can be reused and ports to other manufacturer models and scenarios remain possible.

The project will continue until the beginning of 2020. Until then, the currently available processes and implementations will be refined so that at the end of the project the basis for a product not yet available in this form exists. OPeRAte has the potential to sustainably improve agriculture hopefully not only in Germany through the intelligent application of modern technologies.

ACKNOWLEDGEMENTS

The project is supported by funds of the Federal Ministry of Food and Agriculture (BMEL) based on a decision of the Parliament of the Federal Republic of Germany via the Federal Office for Agriculture and Food (BLE) under the innovation support programme.

REFERENCES

AEF (2019). ISOBUS - AEF Online. URL: https://www.aef-online.org/de/die-aef/isobus.html, accessed: 24.01.19.

Agrawal, H., Prieto, J., Ramos, C., and Corchado, J. M. (2016). Smart feeding in farming through IoT in silos. In Corchado Rodriguez, J. M., Mitra, S., Thampi, S. M., and El-Alfy, E.-S., editors, *Intelligent Systems Technologies and Applications 2016*, Advances in Intelligent Systems and Computing, pages 355–366. Springer International Publishing.

BMEL (2017). Düngeverordnung. URL: https://www.bmel.de/DE/Landwirtschaft/Pflanzenbau/Ackerbau/Texte/Duengung.html, accessed: 28.01.19.

Cavoukian, A. (2009). Privacy by design: The 7 foundation principles. *Information and Privacy Commissioner Ontario, Canada http://www. ipc. on. ca/images/Resources/7foundationalprinciples. pdf Página vigente al*, 22.

DBV (2018). Situationsbericht 2018/19. page 16. URL: https://de.statista.com/statistik/daten/studie/658759/umfrage/wirtschaftskennzahlen-der-landwirtschaft-in-deutschland/, accessed: 28.01.19.

DFKI (2017). SDSD - Smarte Daten, Smarte Dienste. Landwirtschaftliche Datendrehscheibe für effiziente ressourcenschonende Prozesse. URL: https://www.dfki.de/web/forschung/projekte-publikationen/projekte/projekt/sdsd/, accessed: 24.01.19.

Dierks, T. and Rescorla, E. (2008). The transport layer security (tls) protocol version 1.2. Technical report. URL: https://www.ietf.org/rfc/rfc5246.txt, accessed: 30.01.19.

Eclipse Foundation (2019). Eclipse Mosquitto. URL: http://mosquitto.org/, accessed: 13.03.19.

ENISA (2014). Privacy and data protection by design – from policy to engineering. URL: https://www.enisa.europa.eu/ publications/privacy-and-data-protection-by-design, accessed: 25.01.19.

Ferguson, S. (2018). John Deere Bets the Farm on AI, IoT. URL: https://www.lightreading.com/enterprise-cloud/machine-learning-and-ai/john-deere-bets-the-farm-on-ai-iot/a/d-id/741284, accessed: 30.01.19.

Freye, D. (2010). Agrarwirtschaft - KOMOBAR heißt der neue Forschungsschwerpunkt der Hochschule. (11):28 – 29.

HiveMQ (2019). HiveMQ. URL: https://www.hivemq.com/, accessed: 13.03.19.

ISO (2017). ISO 11783-1:2017. URL: http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/05/75/57556.html, accessed: 30.01.19.

Kamilaris, A., Gao, F., Prenafeta-Boldu, F. X., and Ali, M. I. (2016). Agri-IoT: A semantic framework for Internet of Things-enabled smart farming applications. In *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pages 442–447.

Minwoo Ryu, Jaeseok Yun, Ting Miao, Il-Yeup Ahn, Sung-Chan Choi, and Jaeho Kim (2015). Design and implementation of a connected farm for smart farming system. In *2015 IEEE SENSORS*, pages 1–4, Busan. IEEE.

Nordemann, F., Kraatz, F., Tapken, D. H., and Tönjes, D. R. (2016). Ein modulares Framework zur Modellierung, Konfiguration und Regelung von kooperativen Agrarprozessen.

OASIS (2014). Mqtt version 3.1. 1. 1. URL: http://docs. oasis-open. org/mqtt/mqtt/v3, accessed: 30.01.19.

Schlingmann, N. (2016). Communications Networking on Agricultural Machinery.

United Nations, Department of Economic and Social Affairs, Population Division (2017). World Population Prospects: The 2017 Revision, DVD Edition.

Yokotani, T. and Sasaki, Y. (2016). Comparison with HTTP and MQTT on required network resources for IoT. In *2016 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*.