An Architecture Framework for Smart Farming

René Kok
rene.kok2@student.uva.nl

August 31, 2023, 49 pages

Academic supervisor: Dr. ir. M.C. Bor, m.c.bor@uva.nl
Daily supervisor: Martin van de Goor, Martin.vandeGoor@infosupport.com
Host organisation: Info Support, www.infosupport.com

Universiteit van Amsterdam
Faculteit der Natuurwetenschappen, Wiskunde en Informatica
Master Software Engineering

Abstract

Smart farming is an innovative approach to agriculture that integrates information technology and data-driven techniques to optimize farm management practices. This technology can enhance productivity, sustainability, and economic viability in agricultural operations by enabling farmers to tailor their actions to specific conditions. However, current Farm Management Information Systems (FMIS) present significant shortcomings, such as a lack of integration, non-standardized data formats, and interoperability issues.

This research proposes a Smart Farming Architecture Model Framework to address these challenges. The framework is designed to visually represent smart farming systems, making them more accessible and understandable. It can help identify gaps in standardization and serve as a guide for future FMIS development. The framework was validated by mapping two existing use cases from Agrifac, demonstrating its capability to represent diverse elements and interoperable layers within smart farming systems. The framework provides a holistic "system of systems" perspective while drawing inspiration from established models like the SGAM and benefiting from insights from the ISA 95 model and previous research on agricultural architecture frameworks.

Contents

Chapter 1

Introduction

Smart farming is an approach to farm management that utilizes information technology and Data-driven techniques to optimize agriculture practices. These technologies include GPS guidance, data analytics, Remote sensing, and Internet of Things (IoT) devices such as soil moisture sensors, drones, and autonomous tractors. The primary goal of smart farming is to enhance productivity, efficiency, and sustainability in agricultural operations.

By precisely monitoring and controlling factors such as soil conditions, Fertilizer application, Pest infestations, crop health, and Livestock welfare, farmers can optimize their resources and minimize waste. This approach enables farmers to tailor their actions to the specific needs of different areas or even specific plants or animals within a field or herd rather than treating the entire field uniformly. This contrasts traditional farming methods, often involving more generalized approaches to planting, fertilizing, and pest and disease management.

Smart farming can contribute to (long-term) sustainability production and minimize environmental impact. Using site-specific knowledge, smart farming can limit the use of fertilizers and pesticides only where needed [1]. Pesticides can have unintended consequences on non-target organisms, including beneficial insects, birds, and aquatic life [2]. Also, frequent and indiscriminate pesticide use can lead to pest resistance, making them less susceptible to control methods [3]. Unbalanced Fertilizer application can result in water pollution by runoff rain or leaching [4], or can cause soil quality degradation [5], which can cause yield decline.

Smart farming can also increase efficiency. Farmers can make more informed decisions based on real-time data through Remote sensing and Data-driven techniques-driven decision-making. This allows farmers only to apply the resources where they are needed and minimize the usage of limited resources such as freshwater [6]. Agriculture currently accounts for approximately 70% of global freshwater abstraction, utilized to irrigate around 25% of the world's croplands [6]. As the demand for freshwater continues to increase [6], focusing on minimizing freshwater usage in agricultural practices becomes imperative.

Finally, smart farming is also a good economic choice. Through better resource management and more accurate tracking of crop health and quality, yield can be improved [7], resulting in a more enhanced income stream for farmers.

1.1 Problem Statement

While smart farming has the potential to revolutionize farming practices, enhancing productivity, sustainability, and economic viability, the path toward fully embracing this technology is not without obstacles. Current agricultural data management systems, known as Farm Management Information Systemss (FMISs), demonstrate significant shortcomings.

Existing FMISs often tend to be specialized, focusing on one specific task on the farm. This specialization leads to incomplete systems, compelling farmers to use multiple FMISs to meet their diverse needs [8, 9]. Further, these systems and their components lack seamless integration [8], leading to problems with interchangeability between applications and platforms. Compounding these challenges is the high degree of heterogeneity in agricultural systems. The variability of farm products, farming practices, environmental factors, and interrelated objects significantly contributes to the complexity of implementing FMISs.

Another issue within current farming management systems lies in the non-standardized data formats used across different systems and components [8, 9]. This lack of standardization significantly hinders interoperability between various systems, posing severe data communication and integration challenges. The problem is further compounded by the diverse communication protocols, interfaces, and closed and proprietary data formats used by different systems.

Creating an Architecture modeling framework can help to solve the interoperability problems by providing a standardized framework for the discussion, design, and documentation of smart farming systems. This framework will be designed to identify gaps in existing and future standardization. It provides a framework for identifying individual components, data exchanged, and interfaces of the system's landscape. Providing a visual framework for understanding and discussing smart farming systems can help reduce the complexity of these systems and make it easier for architects and farmers to collaborate, identify gaps in standardization, and find solutions to interoperability issues.

1.1.1 Research Questions

This research proposes a smart farming Architecture modeling framework. This framework aims to provide a standardized methodology for designing, discussing, and documenting smart farming systems, thereby addressing interoperability issues. Based on this, the following research question (RQ) was established:

RQ How can an architecture framework be designed to support the development of interoperable smart farming systems?

1.1.2 Research Methodology

We start with identifying and defining the specific requirements of the smart farming framework.

Then we conduct a literature review on existing architecture frameworks aimed at interoperability and smart farming. We are noting features that could be beneficial to incorporate into our framework and recognizing their limitations to overcome.

Next, we will develop the proposed smart farming architecture framework using insights from the problem definition statement, requirement identification, and literature review. This involves iterative refinement, where the initial draft of the framework is continuously improved through feedback from a software architect and domain expert.

Once the framework is finalized, we will conduct validation by mapping the use cases of two already implemented smart farming systems. This validation should test the completeness of our framework and validate if it adheres to the specified requirements.

Finally, the results from the validation process are analyzed, identifying where our framework succeeded and where it fell short. Based on these findings, suggestions are made for improvements to refine the framework further.

Our framework's validation process will take place at Agrifac, a client of Info Support, and will be conducted through the implementation of two specific use cases. Agrifac, which produces field sprayers, utilizes smart farming to optimize field spraying operations. This practical setting at Agrifac will offer a valuable real-world context for the validation of our framework.

1.2 Contributions

An architecture modeling framework designed for smart farming systems can contribute to the field of smart farming. Here are the key contributions:

- It can visually represent smart farming systems, making it easier for architects and farmers to comprehend and work with. By simplifying complex systems, this framework can enhance usability and accessibility, ensuring that technology benefits even non-tech-savvy farmers.
- With the help of our framework, gaps in existing and future standardization can be identified. This can help address non-standardized data formats, ensuring more seamless data communication and integration.
- It can serve as a guide for future development of FMIS, offering clarity on individual components, data exchange protocols, and interfaces.
- It can serve as a common language for stakeholder discussion, promoting better collaboration between system architects, farmers, and other stakeholders.

1.3 Outline

The next chapter, chapter 2, provides foundational information. Topics include smart farming, architecture frameworks, and a discussion about interoperability, setting the stage for the research. Next, chapter 3 investigates previous work in the field. It gives an introduction to the Smart Grid Architecture Model (SGAM) and dissects its framework, objectives, and applications across sectors. Other related models, such as RAMI 4.0, ISA95, and an architecture framework for IoT-based food and farm systems, are also explored. Chapter 4 presents our developed SFAM framework. The requirements and the main elements of the SFAM framework are explained, including the interoperability layers, the smart farming plane, the domains, the zones, and an overall view of the SFAM framework. Finally, this section discusses how to apply the SFAM framework. Chapter 5 provides two use cases to validate the application of the SFAM framework. Finally, chapter Discussion section examines the potential threats to the validity of this study, and chapter 7 summarises the findings of the research and presents the scope for future work in this field.

Chapter 2

Background

This chapter aims to provide the essential background information required for this thesis.

2.1 Smart Farming

Smart farming refers to the application of modern technology and data-driven approaches to agriculture. Farmers can increase efficiency, reduce costs, and achieve sustainable production by leveraging modern tools and technologies. The farm management information systems are central to this transformation and play a role in integrating and managing these technologies.

This technology-driven shift in farming practices has been largely facilitated by the emergence of farm management information systems (FMIS). FMISs are systems that aid farmers in agriculture by catering to various use cases [8]. By collecting, storing, and meticulously analyzing farm-related data, these systems offer insights that can redefine conventional farming methods and automatically actuate. Farmers can gauge the exact amount of water, fertilizer, or pesticide required for a given patch of land through data gleaned from sensors and satellites (see chapter 5). Similarly, in livestock farming, wearable sensors attached to animals offer a glimpse into their health, location, and reproductive patterns [10]. Yield monitoring is another facet where FMIS shines [11]. By taking into account variables like soil health, weather patterns, and irrigation levels, farmers can not only predict but also optimize their yields. On the administrative side, FMIS aids in managing inventories, and keeping track of equipment, seeds, and other resources [8]. The financial component is also catered to, helping farmers budget, forecast, and oversee the farm's financial health [8]. Furthermore, with the unpredictability of weather, climate forecasting has become an invaluable feature, offering farmers foresight to plan their sowing, irrigation, and harvesting [8].

2.2 System Architecture Frameworks

System architecture refers to the high-level structure of a software system, which provides a blueprint for constructing systems. Like architectural blueprints for buildings, system architecture outlines components, their relationships, and the overall design philosophy, ensuring that the system's requirements are addressed.

As the domain of system development grew, the need for standardized methodologies and practices to construct robust architectures became evident. This led to the evolution of system architecture frameworks. These frameworks provide a structured approach, leveraging best practices, guidelines, and tools to create, evaluate, discuss, and implement system architectures.

Instead of representing architecture in a single diagram, the architecture is usually depicted across multiple architectural views. Each view addresses specific stakeholders' concerns, as highlighted by Clements *et al.*[12]. An architecture view showcases a set of system elements and their inter-relations tailored to a particular concern. The utility of having multiple views is to segregate these concerns, providing the modeling, comprehension, communication, and analysis of both software architecture and the related business processes for stakeholders. Importantly, each architecture view is specially defined for a system and must adhere to viewpoints. These viewpoints embody conventions for creating and utilizing a view.

ISO/IEC/IEEE further defines an architecture framework as "Conventions, principles, and practices for the description of architectures established within a specific domain of application and community of stakeholders [13].”

2.3 Interoperability

In system design, interoperability refers to the capability of diverse systems and organizations to work seamlessly together, exchanging and utilizing information coherently and accurately. This principle is especially pertinent when considering integrated systems across various domains, from healthcare to finance and smart farming. Interoperability becomes indispensable as it integrates multiple IT solutions, ranging from Internet of Things (IoT) devices, cloud computing, and data analytics platforms to mobile applications. These components, originating from different manufacturers and developers, must communicate effectively to form a cohesive and efficient system. The lack of interoperability can lead to siloed data, missed opportunities for insights, and suboptimal operational efficiencies.

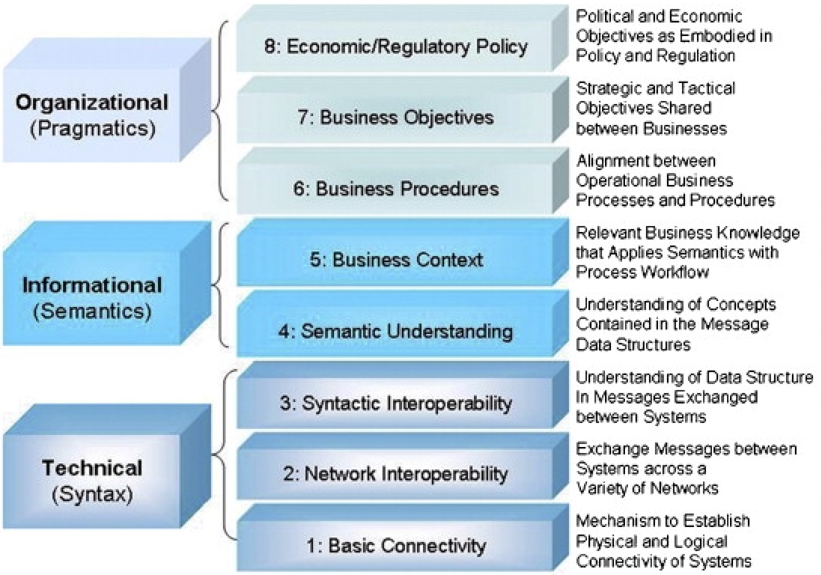

Figure 2.1: Eight categories of system interoperability [14]

In order to achieve interoperability, it is essential to establish a shared understanding across various dimensions - extending from the foundational layers of technology to the broader policies that govern industries and governmental bodies. The GridWise Council proposes eight categories of system interoperability within the electrical grid (figure 2.1) [14]. However, due to their abstract nature, they can be universally applicable to different sectors, including agriculture.

Interoperability, in a technical context, revolves around three primary dimensions: basic connectivity, network interoperability, and syntactic interoperability. Basic connectivity is about the foundational digital exchange of data between systems, ensuring that data can travel between them. This includes standards like Ethernet and WiFi. Network interoperability pertains to how information is transported across multiple communication networks, referencing protocols like FTP and TCP. Syntactic interoperability zeroes in on the rules governing the structure and encoding of messages exchanged, involving formats such as JSON and XML.

From an informational perspective, interoperability is rooted in semantic understanding and business context. Semantic understanding ensures that the concepts and words within messages are comprehended. Business context, on the other hand, brings specificity to information models, refining them according to a particular business process. This contextualization often builds on foundational semantic models while integrating added constraints relevant to specific business operations.

Organizational interoperability dives into the alignment and agreement between business organizations. It’s essential that businesses have compatible procedures and mutually beneficial objectives to interact effectively. For example, a farmer and a consumer might have established protocols for the sale and distribution of goods. Beyond immediate operational goals, there’s also the bigger picture of overall business objectives, which provide a guiding framework for these specific processes. All these interactions are also influenced by the broader landscape of economic and regulatory policies.

Chapter 3

Related Work

In this chapter, we explore the Smart Grid Architecture Model (SGAM), a framework for architecting interoperable smart grid systems. We'll discuss RAMI 4.0, an industry-centric application of SGAM. Additionally, we'll also examine research on smart farming system architecture framework and delve into the ISA95 categories to distinguish between physical processes and information systems. Together, these insights lay the foundation for our architectural framework.

3.1 An Introduction to the Smart Grid Architecture Model

The Smart Grid Architecture Model (SGAM) is a proven functional implementation of an architecture framework to document and describe smart grid system architectures. It is a structured approach to describe the various aspects of smart grids, including the communication technology, functions, and processes involved in delivering efficient, reliable, and sustainable electricity.

The significance of SGAM stems from the need to manage the increasing complexity of modern energy systems. As the integration of renewable energy sources, energy storage systems, electric vehicles, and advanced metering infrastructure increases, the grid's operations become more complex [15]. SGAM allows stakeholders to visualize and manage this complexity in a structured way, facilitating decision-making and system design processes [15].

3.1.1 Understanding SGAM's Core Objectives

The primary goal of SGAM is to enhance interoperability within the smart grid, ensuring that distinct components, regardless of their manufacturer or provider, can work together seamlessly. It plays a crucial role in promoting standardization, as it aids in creating standardized interfaces and data models. This reduces the complexity of integrating different components and systems within the grid, thus promoting system efficiency and reliability. SGAM visually represents the smart grid's operational intricacies, enabling stakeholders to grasp and manage system complexity more effectively. It also provides a framework for conceptualizing, designing, and deploying smart grid technologies, thereby streamlining the process of incorporating new technologies and strategies into the grid.

Moreover, SGAM is used to analyze and validate smart grid use cases and architectures. This approach ensures that proposed designs meet the grid's requirements and can be integrated effectively, enhancing the grid's functionality and future-proofing energy infrastructure. By achieving these objectives, the SGAM contributes significantly to the modernization and sustainability of our power systems.

3.1.2 Dissecting the SGAM Framework

The SGAM model has three axes; an axis for the interoperability dimensions, one for the domains, and one for the hierarchical zones. The interoperability dimension contains four layers: Business, Function, Information, and Communication. These axes serve as the foundation for designing interoperable smart grid systems.

The domains represent the complete electrical energy conversion chain, including Generation, Transmission, Distribution, Distributed Energy Resource (DER), and Customer Premises. Each domain focuses on a specific stage of the energy conversion process.

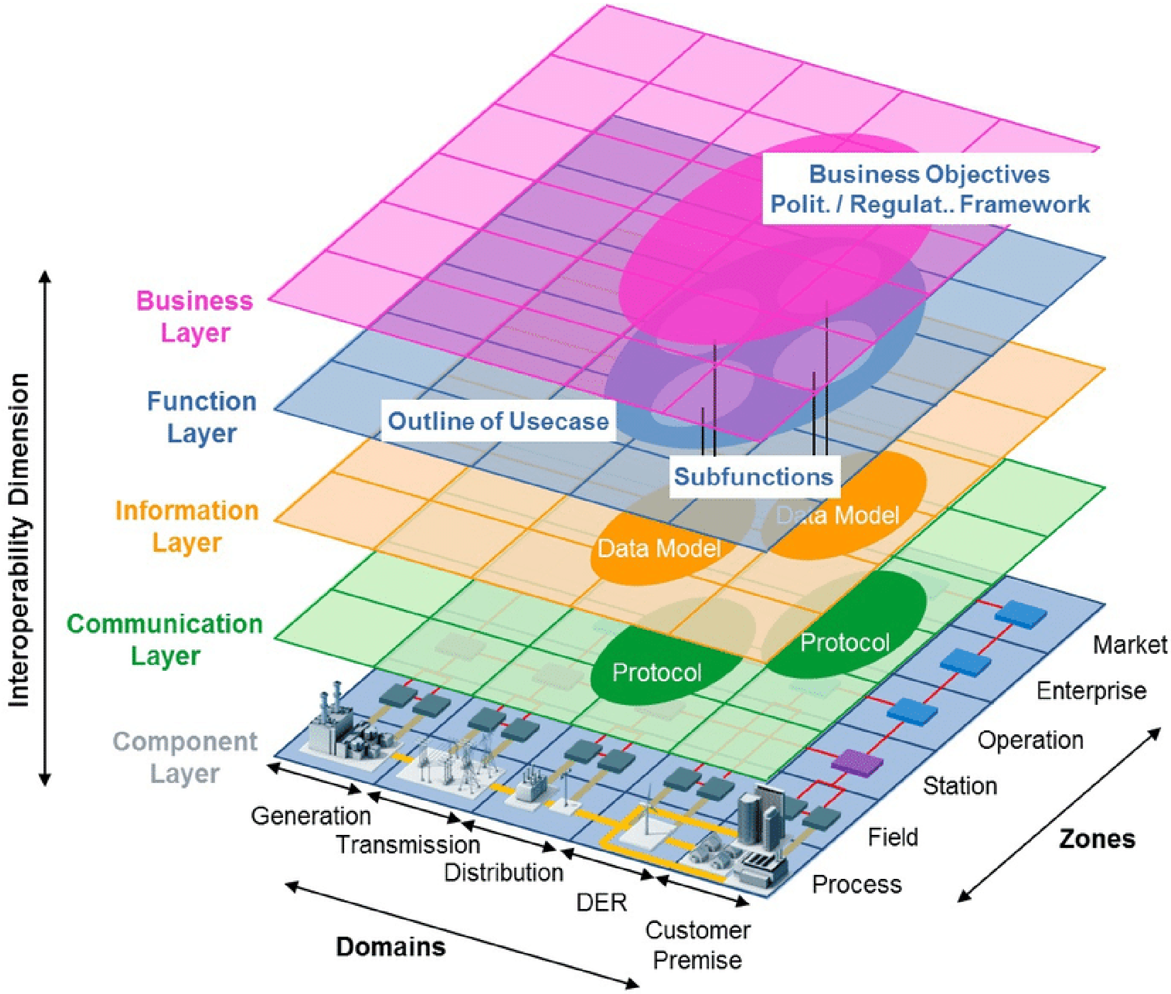

Figure 3.1: Visualization of the Smart Grid Architecture Model (SGAM), showcasing its three axes

The (hierarchical) zones represent the Information and Communication Technology (Information and Communication Technology (ICT)) based control systems that manage the energy conversion chain. The zones are Process, Field, Station, Operation, Enterprise, and Market. These zones are modeled after the Purdue Reference Model for computer-integrated manufacturing, as established in the IEC 62264-1 standard for enterprise-control system integration [16]. The SGAM zones represent a hierarchical structure in power system management.

Overall, the goal of defining these layers, domains, and zones is to establish the multi-layered, interconnected nature of the modern energy system, from the physical processes of energy conversion to the ICT-based systems that control and manage them. By utilizing these axes, a comprehensive topological map can be architected that showcases a smart grid system's diverse components and interdependencies.

3.1.3 SGAM's Versatility: Applications Across Multiple Sectors

The SGAM framework has been applied in various sectors, from electric mobility to smart cities, maritime framework, and Industrie 4.0. Gottschalk *et al.* presents a few of these applications [17].

The Electric Mobility Architecture Model (EMAM) is a high-level model developed to illustrate the relationships between components in the electric mobility sector. It incorporates the same layer concept as the SGAM and focuses on integrating electric vehicles into the Smart Grid, detailing the connections from electric cars to power grid.

Another application of the SGAM is seen in the Smart City Infrastructure Architecture Model (SCIAM). This model is a derivative of the SGAM and was proposed in the German DIN/DKE Smart Grid Standardization roadmap for Smart Cities. It introduces an action layer replacing the business layer in the SGAM, a concept yet to be agreed upon. The SCIAM also uses a new axis to describe functionalities in a smart city, affecting multiple domains. These domains include Supply/Waste Management, Water/Waste Water, Mobility Transport, Healthcare/AAL, Civil Security, Energy, Building, and Industry. The SCIAM is designed to illustrate various scenarios, like the application of smart traffic lights or industrial waste management, indicating information flow across multiple domains.

The Maritime Architecture Framework (MAF) is yet another adaptation of the SGAM in the maritime sector. It was developed as a framework for displaying the information exchange between maritime actors and services from a Maritime Cloud, which provides standardized protocols for a host of services. The information exchange between the Maritime Cloud and additional actors in the maritime sector is developed to be interoperable.

Finally, the Reference Architecture Model for Industrie 4.0 (RAMI 4.0) represents a more complex derivative of the SGAM framework. It is based on the German Industrie 4.0 concept and introduces a new Integration layer to the existing SGAM framework. This layer represents components or systems without ICT interfaces, displaying them as a virtual representation in the digital value chain.

Each model continues to expand on the original principles of the SGAM, demonstrating its broad applicability and flexibility in various domains.

3.1.4 Applying SGAM to Smart Farming

Although the SGAM framework has been successfully adapted to various sectors, its application in farming remains unexplored. The adaptability of this framework could help address the lack of integration between different FMIS. An SGAM-derived Smart Farming Architecture Model (SFAM) could facilitate better integration between systems and components, addressing one of the critical issues identified by Tummers *et al.* [8]. It could provide a comprehensive framework for integrating various systems, from crop management to livestock tracking, irrigation systems to weather monitoring, all under a unified architecture model. The SGAM also promotes standardization in data formats and interfaces. This could prove invaluable for FMIS, where the lack of standardized data formats is causing problems. SFAM could propose and promote standardized data formats, making data exchange and integration more straightforward and effective.

3.2 RAMI 4.0

The RAMI 4.0 framework provides an additional "Connected World" hierarchical zone. It represents a crucial area of the model, which focuses on the communication and interaction between systems in different locations. This layer underscores the Industry 4.0 principle of a hyper-connected world where data from various sources is integrated, analyzed, and used for decision-making [18].

It encompasses global networks, remote systems, and any communication that occurs between the assets located in these various sites. These might include machines in different factories, supply chain networks, or any Internet of Things (IoT) devices that are geographically dispersed. The connected world layer allows different systems to work together seamlessly.

The integration and sharing of agricultural data with third-party systems - including weather forecasting systems, governmental bodies, research groups, and customers - can significantly amplify the efficiency and productivity of agricultural systems. Exchanging real-time weather data can enable proactive adjustments to farming strategies. Interfacing with governmental bodies can inform more pertinent agricultural policies. Collaborative efforts with research groups can stimulate innovation in farming practices.

3.3 ISA95

Originally known as S95, ISA-95 is a widely recognized framework within the international production industry, spanning pharmaceuticals, petrochemicals, and food processing sectors. The model is fundamental in integrating office automation with production automation and mechanization, focusing on three core aspects: the exchange of information between enterprise management systems and manufacturing operations systems, the activities within manufacturing operations systems, and the information exchanged within these systems. Based on the Purdue Reference Model, which forms the basis for the hierarchical structure of the SGAM model [16], the ISA-95 framework establishes five control levels [19]:

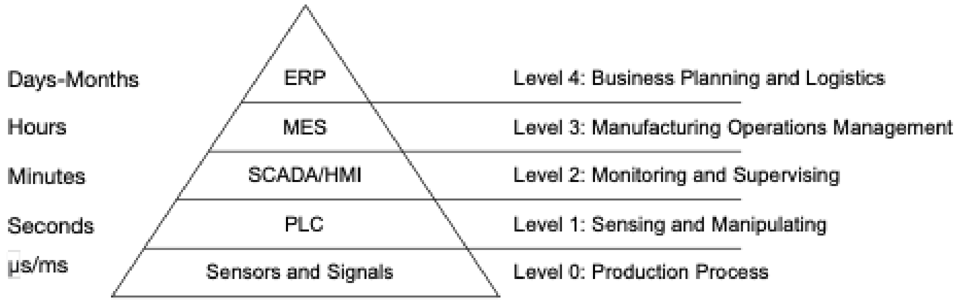

Figure 3.2: The ISA-95 pyramid showing the different control levels [20]

Level 0 The Process: This level represents the physical processes. It could involve machines, actuators, sensors, and other physical components. At this level, measurements and control functions occur.

Level 1 Basic Control: This level directly controls the Level 0 physical processes. It involves equipment like sensors, controllers, and actuators that gather data about the process and enact direct control measures based on that data. It could be real-time automated control systems like programmable logic controllers (PLCs).

Level 2 Supervisory Control: This level manages the Level 1 basic control systems. It typically involves supervisory control and data acquisition (SCADA) systems, distributed control systems (DCS), or other industrial control systems that monitor and coordinate the activities of Level 1 controllers. They are responsible for setting set points, controlling loops, and collecting data for longer-term analysis and reporting.

Level 3 Manufacturing Operations Management: This level is concerned with managing operations on a broader scale. It includes systems like Manufacturing Execution Systems (MES) or Manufacturing Operations Management (MOM) systems. They manage production scheduling, resource management, order tracking, material tracking, and more. They typically operate on a time scale of shifts or hours.

Level 4 Business Planning and Logistics: This is the enterprise-level, often interacting with or encompassing ERP (Enterprise Resource Planning) systems. These systems deal with business activities like demand forecasting, production planning and scheduling, inventory management, procurement, and distribution. The time scale of Level 4 systems can range from days to months or even years.

3.4 Architecture Framework of IoT-based Food and Farm Systems

A contribution in the context of architecture frameworks for smart farming was made by Verdouw *et al.* in their research on the interoperable challenges faced when implementing IoT in the agricultural domain [19]. They propose an architecture framework for modeling IoT-based systems within the agriculture and food sectors. It encompasses a systematic set of architectural viewpoints and guidelines for their practical application. This approach allows them to model individual IoT systems' architectures.

Their research was part of the European IoF2020 project, where the framework's applicability and effectiveness were tested across a diverse set of agricultural sub-sectors and various supply chain roles. The universality of their framework was demonstrated as it proved instrumental in serving as a shared language that facilitated the alignment of system architectures across various autonomous IoT-based systems in agriculture and food.

They introduce a business process hierarchy view which provides an overview of business processes and how they relate to each other. This view consists of hierarchical levels, which are primarily inspired by the ISA-95 reference model (see section 3.3). These hierarchical levels are similar to SGAM's hierarchical zones.

However, while the paper envisions IoT-based systems functioning as interoperable but autonomous components within a more extensive system of systems, it hasn't explicitly addressed this perspective. More research is needed to understand how the architecture framework can effectively support this vision.

Chapter 4

The Smart Farming Architecture Model (SFAM) Framework

The SFAM framework has been carefully crafted in consultation with a domain expert and software architect to provide a structured approach to designing smart farming systems. It offers an architectural perspective that is both specific and technology-neutral. Aligned with the requirements, the SFAM framework enables the validation of current and new smart farming systems and promotes the design of interoperable systems.

4.1 Requirements for the SFAM Framework

This section presents the Smart Farming Architectural Model requirements derived from the problem statement discussed in section 1.1 and the background information provided in chapter 2.

4.1.1 Interoperability Categories Compliance

Given the interoperability problems that currently plague FMISs, one of the critical requirements for the Smart Farming Architectural Model is to comply with interoperability categories. The model should incorporate elements that enhance syntactic interoperability, such as standardized data formats and protocols, but also the various assets and machinery. It should also foster semantic interoperability by ensuring data meanings and business contexts are preserved and understood across systems.

Finally, to address organizational interoperability, the model must be flexible enough to accommodate varying organizational processes inherent in diverse farming practices.

4.1.2 Supporting Diverse FMIS Use Cases

Given the vast array of FMIS use cases, ranging from crop management to livestock health monitoring, the architectural model should be flexible and scalable enough to support these diverse scenarios.

The architectural framework must cater to various FMIS applications, such as financial management, farm asset management, crop monitoring, and livestock management. Therefore, the model should be designed to accommodate these use cases, allowing for easy integration, configuration, and customization.

4.1.3 System Mapping Capabilities

The capability to map existing and new systems is another essential requirement for the Smart Farming Architectural Model. This entails having a model that can accurately represent the current state of the farm's digital ecosystem and anticipate future additions or modifications.

4.2 SFAM Framework Elements

The SFAM framework encompasses multiple dimensions for designing and organizing smart farming systems. SFAM incorporates layers, domains, and hierarchical zones.

The layers in SFAM provide a structured representation of different aspects of the system. These layers include the business layer, which addresses business objectives and processes; the functional layer, which describes system functionalities; the information layer, which focuses on data exchange and models; the communication layer, which deals with communication protocols and components; the integration layer, which links various components and facilitates interoperability; and the asset layer, which plays a crucial role in capturing and managing the physical resources and digital representations essential to smart farming systems.

The domains in SFAM cover different areas of agricultural activities, such as cultivation area, livestock, storage, processing, and distribution.

The hierarchical zones in SFAM represent levels of agricultural business processes, including the connected world, management information, operation execution, production control, sensing and actuation, and physical objects. By integrating these dimensions, SFAM provides a comprehensive framework for designing and managing smart farming systems, enabling interoperability and supporting future scenarios.

4.2.1 SFAM Interoperability Layers

Similar to precision agriculture, smart grids involve diverse technologies and disciplines, requiring interoperability to integrate these components for comprehensive, cross-disciplinary solutions. In the Smart Grid Architecture Model, the layers provide a structured way of modeling different aspects of the respective systems. They describe the various elements that make up the systems and how they interact. They offer a holistic view that spans from high-level business processes to specific technical details regarding data transmission and processing. SGAM aims to guarantee interoperability and efficiency at every system level. However, as SGAM's design primarily caters to the electric grid, alternate layer designs may be more suitable and relevant in the context of precision agriculture.

The Reference Architecture Model Industry 4.0 (RAMI 4.0) presents interoperability layers that closely resonate with precision agriculture's technological and operational requirements. Grounded in the SGAM methodology, RAMI 4.0 encapsulates critical concepts such as the Internet of Things (IoT), cyber-physical systems, data analytics, and edge and fog computing [18, 21] — all significantly harnessed in precision agriculture.

One distinct feature of RAMI 4.0 is the introduction of new layers that extend beyond SGAM's framework. RAMI 4.0 includes an 'Asset' layer and an 'Integration' layer [17]. The Asset layer represents the component layer in the SGAM framework but focuses on managing physical assets such as tractors, farming equipment, and drones, which are essential in precision agriculture. On the other hand, the Integration layer represents physical reality, containing objects such as processes, machines, sensors, actuators, and documentation [22]. Incorporating these additional layers significantly amplifies the suitability of RAMI 4.0's interoperability layers for designing precision agriculture applications.
Based on RAMI's interoperability layers, we present the following SFAM interoperability layers in figure 4.1.

Business Layer

This layer addresses the business-related aspects of a precision agriculture system, such as business processes, market strategies, resource planning, and customer relationship management. It is concerned with ensuring that the system aligns with the overall objectives of the farming business, whether that is increased yield, better crop quality, improved efficiency, or a combination of these and other goals.

Functional Layer

The functional layer describes the functionalities provided by the system, such as automation, decision support, data analysis, and others. It outlines the roles, capabilities, and responsibilities of the different components of the system. In precision agriculture, this might include GPS guidance for machinery or analytics tools for processing and interpreting data.

Information Layer

The information layer outlines shared and utilized data among system functions, services, and components and includes information objects and standardized data models. These objects and data models

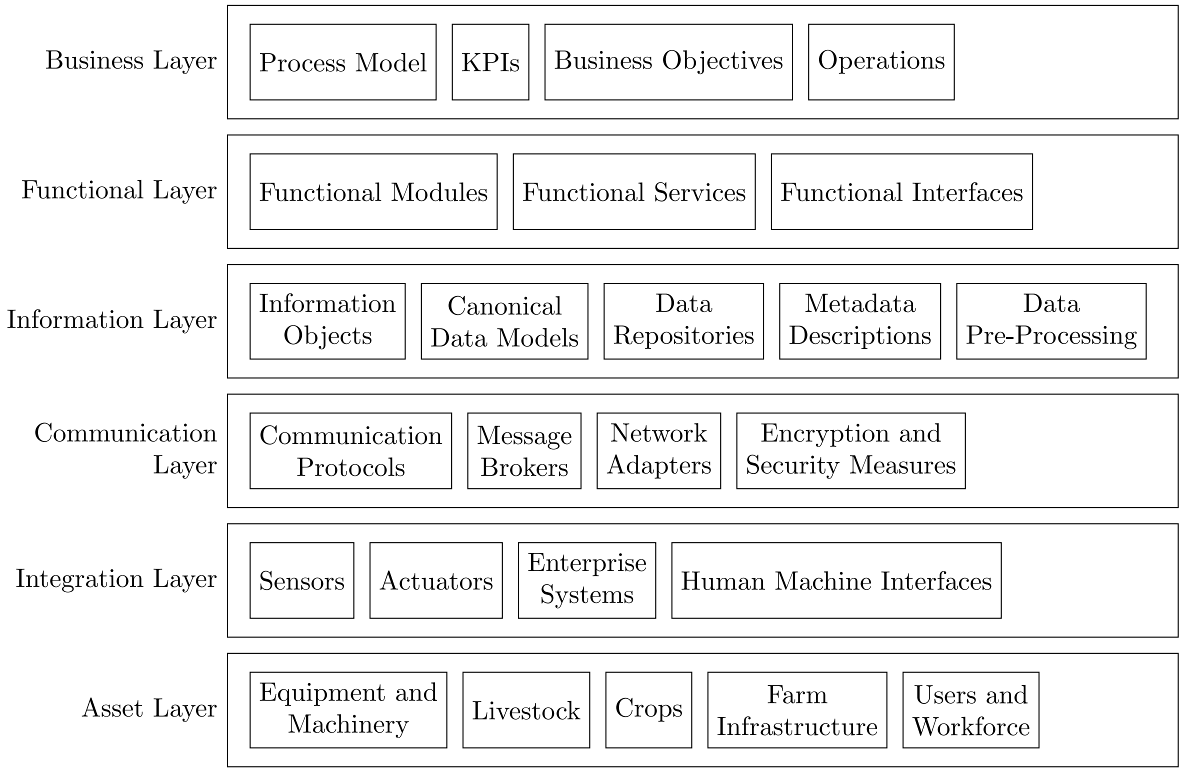

Figure 4.1: SFAM Interoperability layers

embody the standard semantics for functions and services, enabling an interoperable exchange of information. The information layer encapsulates data storage and pre-processing in the edge, fog, and cloud computational environments.

Communication Layer

This layer deals with transmitting information between devices and systems. It includes the networks and protocols used for this communication. In the context of precision agriculture, this might involve wireless communication between field sensors, machinery, and the farm's central management system. It might also involve using internet-based services for data storage, analytics, and other purposes.

Integration Layer

This layer links various components and personnel with the agricultural information systems. This level includes sensors and actuators (installed on farm equipment), along with enterprise-level systems like Manufacturing Execution Systems (MES) and Enterprise Resource Planning (ERP). It also comprises interfaces designed for human interaction, allowing users to interact directly with the agricultural platform and broader enterprise systems.

Asset Layer

The asset layer represents the physical and virtual objects, systems, and components utilized in a precision agriculture context. This could include physical assets like tractors, drones, irrigation systems, sensors, and more, as well as digital counterparts of physical entities - the so-called "digital twins" - which may represent machinery, crop species, or livestock.

4.2.2 SFAM - Smart Farming Plane

In the same vein as with smart grid systems, Precision Agriculture systems distinguish between the physical process and information management perspectives. These perspectives can be segmented into the physical domains of the agricultural production chain and hierarchical zones (or levels) for managing farming processes.

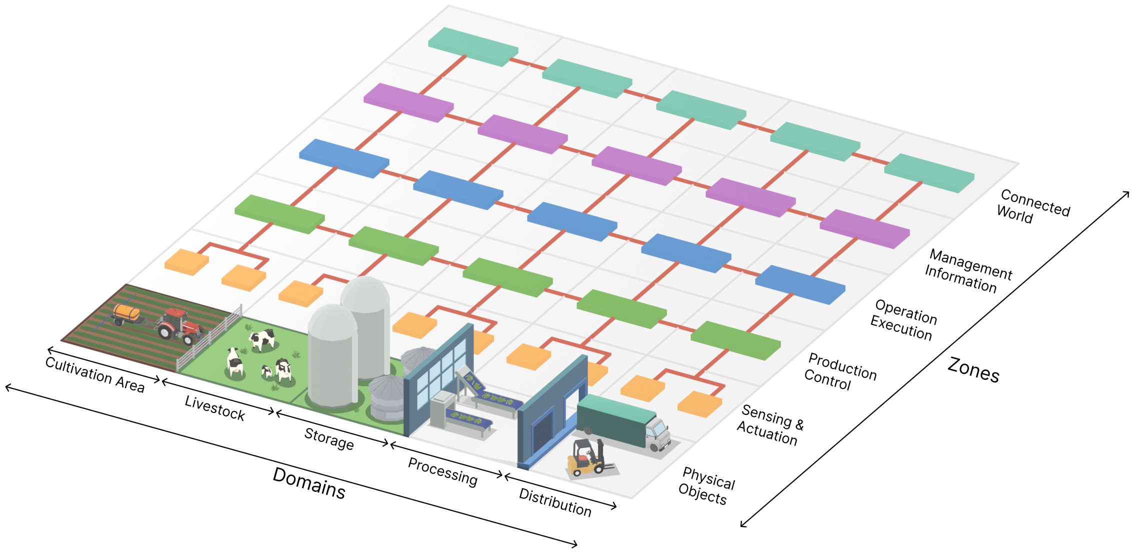

Figure 4.2: The Smart Farming Plane with their domains and hierarchical zones

Applying this concept to the Precision Agriculture model provides the foundation for the Smart Farming Plane (see figure 4.2). This smart farming plane facilitates the representation on which levels (hierarchical zones) of agricultural management interactions between domains occur.

4.2.3 Domains

Just as the SGAM Plane encompasses the entire electrical energy conversion process, we aim to create an equivalent comprehensive framework for the precision agriculture value chain. However, it is essential to note that the SGAM domains are not directly applicable to smart farming. Therefore, we will establish our unique domains tailored to PA. These domains must cover the on-farm activities so that the architecture model can give a comprehensive overview. We extracted the domains based on the agricultural value chain [23]. To ensure the versatility of our framework, we made a conscious effort to maintain the abstract nature of the domains, allowing it to accommodate a wide range of activities across various agricultural sub-sectors. This includes but is not limited to horticulture, livestock, agroforestry, and aquaculture.

Cultivation Area

This domain is concerned with the area where the actual farming happens, which includes soil and climate management. In smart farming, technology can be used to monitor soil health, track weather conditions, and automate irrigation systems. It encompasses everything related to crop growth, including pest and weed management and the use of data analytics for crop rotation and field utilization to maximize yield. This domain has many applications, including fields, greenhouses, orchards, and more.

Livestock and Animals

This domain pertains to raising and managing animals for food, fiber, milk, eggs, or other products. The livestock domain might include systems for tracking the health, breeding, and behavior of animals. This could involve wearable devices for animals, automated feeding systems, and even AI-driven systems for detecting signs of disease or stress in animals.

Storage

This domain covers all aspects of storing agricultural products, from the initial harvest to long-term storage. In a smart farming context, it could involve systems for monitoring storage conditions to ensure optimal temperature, humidity, and other factors. This might include using sensors to detect any changes in conditions and automated systems to adjust conditions as necessary.

Processing

The processing domain involves transforming raw agricultural products into food, fiber, and other goods. In smart farming, this might involve automated machinery for sorting, cleaning, and packaging agricultural products. It could also include systems for tracking the quality of processed goods and even using data analytics to optimize processing operations.

Distribution

The distribution domain covers the entire supply chain, from getting agricultural products from the farm to wholesalers or retailers. This could involve everything from logistics management systems to traceability solutions for ensuring food safety and quality.

In our research, we assessed the comprehensiveness of our SFAM framework by mapping 81 features of FMIS. These features were identified from 38 primary studies included in Tummers *et al.*' research on the characteristics and challenges of FMIS [8]. We utilized this mapping technique to evaluate the scope and effectiveness of our SFAM framework. This mapping, detailed in Table A.1 in appendix A, aimed to validate the breadth and applicability of SFAM domains by aligning them with specific FMIS features. The table provides a detailed overview of how FMIS features distribute across the various SFAM domains. It underscores the interoperability of these domains, as many features, such as 'Financial Management,' 'Data Acquisition,' 'Reporting,' and 'Operation Plan Generation,' span multiple domains. Conversely, some features, like 'Calibration Management' and 'Harvest Management,' are exclusive to the 'Cultivation Area' domain.

4.2.4 Zones

The SFAM zones represent the hierarchical levels of agricultural business processes as presented in figure 4.3. The hierarchical levels of agricultural business processes represented by the SFAM zones are clarified through Verdouw *et al.* [19]'s Business Process Hierarchy Viewpoint, which comprises four distinct layers aligned with the ISA-95 reference model, as discussed in chapter 3.3. The viewpoint has been proven effective in the agricultural sector by its application in numerous case studies [24]. Additionally, RAMI 4.0's connected world, as discussed in chapter 3.2, has been borrowed for integration with external systems.

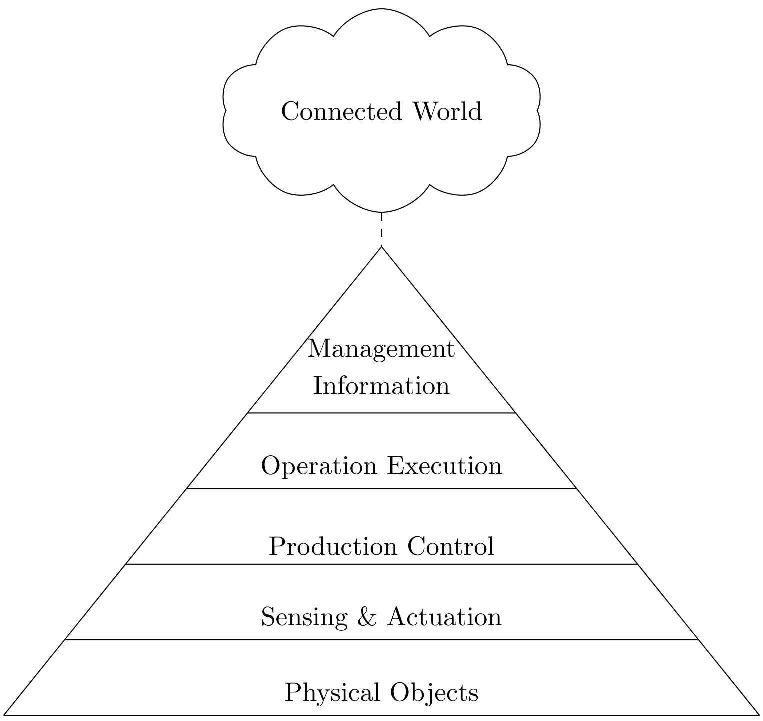

Figure 4.3: SFAM zones

Connected World

The Connected World is the topmost layer in the hierarchy, representing the global connectivity and information exchange between various assets, locations, and systems in different geographical zones. It encapsulates interactions with more extensive networks and global systems, including those beyond the immediate control of the enterprise, such as interactions with external logistics providers, data from global market trends, weather systems, and third-party platforms. This layer can include elements like cloud servers, IoT platforms, remote monitoring systems, and external databases.

Management Information

The Management Information layer deals with business planning and logistics, taking a macro perspective to oversee an entire enterprise's management. Specific responsibilities include production planning and scheduling, efficient material use, shipping logistics, and inventory management. These activities are often carried out using sophisticated tools like Enterprise Resource Planning (ERP) systems, which provide an integrated view of core business processes in real-time. It operates on a large-scale time frame, considering matters spanning months, weeks, or days. This layer is essential in strategic planning and overall direction setting for the business.

Operation Execution

This layer focuses on administering production workflows, whether batch, continuous, or discrete operations. It's chiefly associated with Manufacturing Execution Systems (MES), which track and document the transformation of raw materials into finished goods. This zone operates on a moderate timescale, managing tasks and ensuring the smooth execution of daily operations.

Production Control

Digging deeper into the hierarchy, the Production Control layer manages supervisory, monitoring, and controlling activities of physical processes. It utilizes systems like Supervisory Control and Data Acquisition (SCADA), Programmable Logic Controllers (PLC), and Distributed Control Systems (DCS). This layer operates in real-time, ensuring tasks are carried out swiftly and efficiently. With a focus on the immediate term, it operates within the shortest time spans - from minutes down to seconds and milliseconds.

Sensing & Actuation

This layer is essentially the farm's sensory and responsive mechanism. It involves systems that collect data from the production environment and perform actions based on it. Sensors monitor parameters like temperature, pressure, or flow rate, while actuators respond to this data by adjusting operations, optimizing efficiency, and maintaining safety standards.

Physical Objects

This is the most tangible layer, directly related to the physical entities involved in the business process. It covers many elements, including fields, stables, plants, animals, machinery, processing facilities, containers, boxes, and vehicles. Moreover, it includes human elements like employees or customers. This layer represents the physical world where all the processes and systems above operate.

Direct Layer Communication

There are instances where direct communication between non-adjacent layers can occur, bypassing intermediary layers. This can happen for various reasons, such as the need for faster response to real-time data, addressing specific requirements, or simply optimizing efficiency. In these cases, the information flows directly between the involved layers without going through the traditional hierarchical flow.

For example, the Connected World layer can directly communicate with the Operation Execution layer without involving the Management Information layer. This is particularly useful when the farm operation needs to respond to real-time external data, like weather information. When a weather service in the Connected World detects adverse weather conditions that could affect the farm operation, it could directly alert the Operation Execution layer, which can then adjust the production workflows accordingly without waiting for the Management Information layer to process and relay the information.

4.2.5 SFAM Framework

By merging the interoperability layers described in chapter 4.2.1 with the described smart farming plane, we create our Smart Farming Architecture Model (SFAM). This model spans across three dimensions:

- The Interoperability Layers
- The Domains
- The Hierarchical Zones

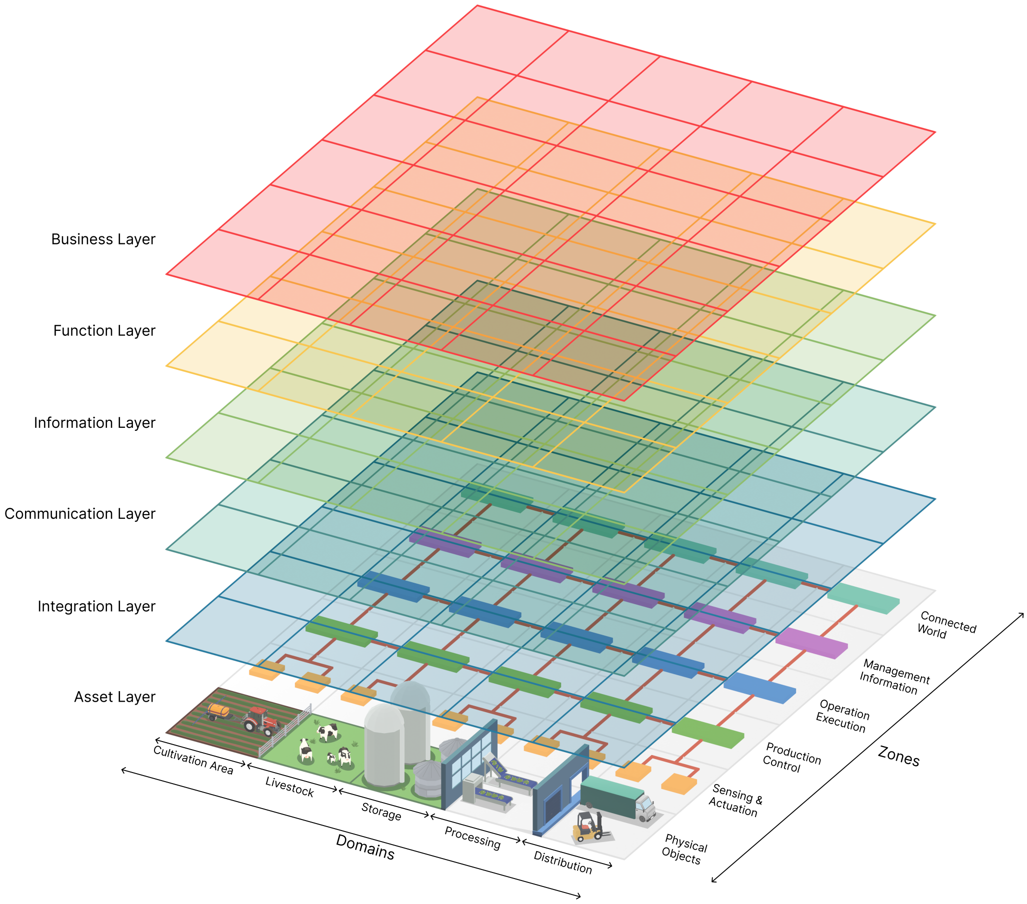

Figure 4.4: The Smart Farming Architecture Model with interoperability layers

The SFAM framework allows the representation of entities and their relationships in the context of smart farming domains and information management hierarchies in consideration of interoperability aspects.

4.3 Applying the SFAM Framework

This chapter delves into a step-by-step approach for applying the SFAM framework. The entire process and its steps are visually represented in figure 4.5. Following these steps, one should be able to iteratively develop documented designs for smart farming systems.

Figure 4.5: Use case mapping process to SFAM

Use Case Analysis

The starting point is a use case analysis. This analysis is a technique used in systems engineering and software development to identify, clarify, and organize system requirements. It provides a method to capture who (actors) does what (use-cases) with the system for what purpose (goals). The central component of a use case analysis is the use case, which describes a system's behavior as it responds to a request that originates from one of the system's actors.

The desired result of a use case analysis is a clear understanding of the system's requirements. These requirements serve as a foundation for system design, testing, and validation. The analysis helps to ensure that the final system meets the needs of its users and integrates well with other systems. Identifying hard constraints and requirements during the use case analysis is crucial, as they impact the development of the following steps.

Development of the Asset Layer

The elements within the asset layer originate from the actors identified in the individual use case. Each actor, whether it's a device, application, person, or organization that engages with the system, is situated within the appropriate domain and zone of the Smart Farming Architecture Model.

Development of the Integration Layer

Having identified the assets, we can now discern the components within the integration layer that should bridge the physical and digital worlds. These components primarily encompass sensors, actuators, and sophisticated, enterprise-grade systems such as Manufacturing Execution Systems (MES) and Enterprise Resource Planning (ERP). Additionally, this layer includes human-centric interfaces that enable users to engage directly with the agricultural system as well as broader enterprise systems.

Development of the Business Layer

Next, the business layer serves as a repository for all business-related documentation. This crucial layer encompasses various components such as business objectives, economic considerations, and regulatory constraints that form the foundation of the given use case. Like the other layers, the business entities are placed within their respective SFAM domains and zones.

Development of the Function Layer

We can derive the system's functionalities after identifying the business-related components within the use case. A typical use case is comprised of multiple sub-use cases. By abstractly conceptualizing these sub-use cases, isolated from specific actors, they can be converted into functions. These functions can then be appropriately assigned to each component outlined within the integration layer.

Development of the Information Layer

The development of the information layer commences once the functions have been identified and defined. This layer is a structured blueprint for sharing and using data throughout the system. Special attention must be given to where data will be stored and pre-processed within the system. Depending on the system's needs, this could occur in edge, fog, or cloud computational environments. The information objects and data models are assigned to the appropriate domain and zone. All these considerations work together to ensure that the information layer supports robust data usage, enhances overall system interoperability, and contributes to the system's ability to fulfill its use case requirements.

Development of the Communication Layer

Finally, the communication layer is designed to ensure seamless transmission of information between components. In developing this layer, the key step is identifying the networks and protocols to facilitate communication within the system. The selection of suitable protocols and mechanisms is based on the information objects, canonical data models, and the consideration of the use case's non-functional requirements. These protocols and mechanisms are then assigned to the corresponding domain and zone in use.

Chapter 5

Application of the SFAM Framework

To validate the SFAM framework presented in the previous chapter, the framework was applied to two use cases through two case studies. The following sections offer an overview of these individual case studies.

The two case studies were conducted at Agrifac with their software architect and information management manager. Agrifac is a Dutch company that manufactures beet harvesters and field sprayers. Agrifac's commitment to precision farming techniques enables them to maximize yield through minimal resource expenditure, including water and pesticides. This is achieved through precise, plant-level dosing strategies. The selected use cases for the study are related to Agrifac's field sprayer, the Condor.

5.1 Use Case Monitor Condor Engine

This use case outlines the application of active monitoring of Agrifac's Condor fieldsprayer engine using sensory technology. The system triggers an auditory alert via an onboard buzzer and visual cues on the Human-Machine Interface (HMI) panel if it detects any anomalies. Further, the case offers offsite monitoring through the My Agrifac portal, a web and mobile application providing services like real-time machinery tracking and a parts store. These features enhance operational efficiency in agriculture. The portal's internet-based connectivity simplifies troubleshooting and maintenance, reducing downtime and boosting productivity.

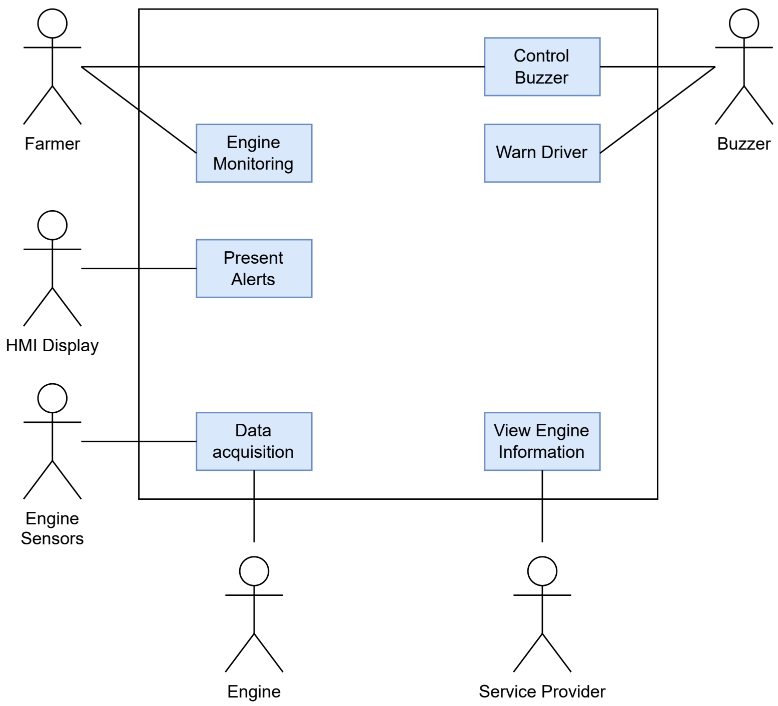

Figure 5.1: Use case diagram for Monitor Condor Engine

The primary objective of this use case is to enhance operational efficiency and minimize downtime of the Condor field sprayer. The strategy entails real-time engine monitoring for early malfunction detection, prompt alerts, and preventing expensive repairs and excessive wear and tear. Another aim is to improve maintenance planning and response times. Service providers can access engine data through the 'My Agrifac' portal, enabling proactive issue resolution, scheduling preventive maintenance, and fast problem-solving.

Actor	**Actor Type**	**Description**	**Manufacturer/Model**
Farmer	End-user	The farmer is an individual who owns or works on a farm, engaging in agricultural activities such as crop cultivation, livestock rearing, and other related tasks. They drive the Condor field sprayer.	-
Service Provider	End-user	The service provider is an individual or organization that offers specialized services related to Agrifac's Condor field sprayers.	-
Condor Field Sprayer	Machinery	Agrifac's Condor field sprayer is an agricultural machinery designed to spray fields with water or pesticides. It features technology for spot spraying and disease detection. It can target individual plants to minimize chemical usage and environmental impact.	Agrifac
HMI Touch Display	System	The HMI Touch Display is a user interface system. It provides a graphical interface that allows users to interact with the Condor's system using touch-based inputs.	Christ Windows 10 Embedded
Engine	Device	The engine is a mechanical device that converts energy into mechanical force and motion. It provides the necessary power to operate the Condor.	Cummins
Buzzer	Device	A buzzer is an electronic device that produces a buzzing or beeping sound. It is used to provide audible alerts.	-

Table 5.1: List of actors 'Monitor Condor Engine'

Development of the Asset Layer

The content of the asset layer is derived from the list of actors (see table 5.1). These actors are placed in their appropriate domain and zone (5.2).

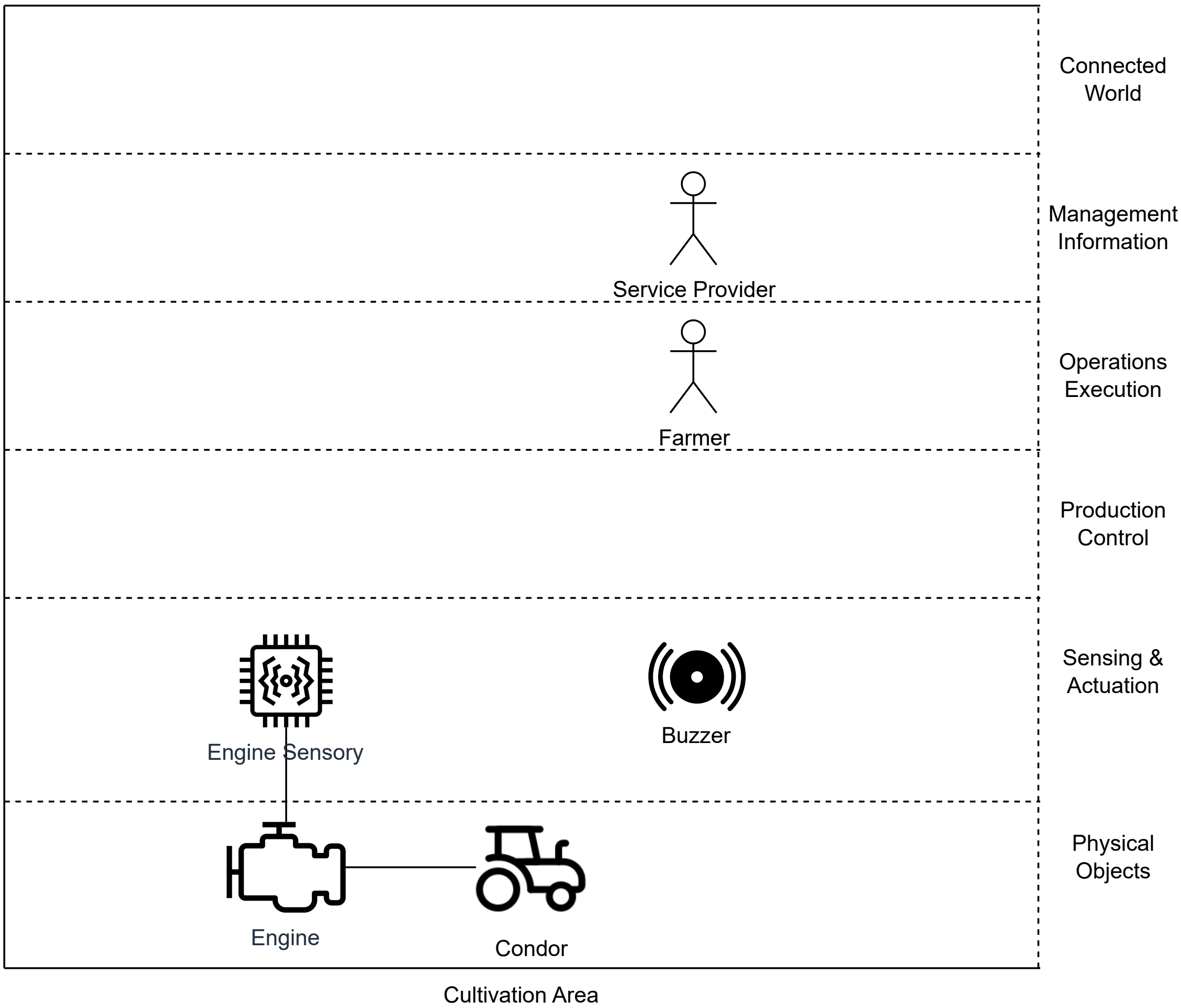

Figure 5.2: Asset layer for 'Monitor Condor Engine'

Development of the Integration Layer

After identifying the asset layer, we must bridge the physical assets with the digital world through the integration layer (5.3).

The exact details of the engine's specifications remain undisclosed; however, it incorporates many sensors. These sensors are connected to one PLC, which acts as a conduit to other systems. An additional PLC reads sensor data from the engine's PLC. This second PLC manages the buzzer and dispatches alert notifications to the HMI panel. These components establish a communication network through CAN bus. Additionally, a router is affixed to the HMI panel, providing a conduit to Microsoft's Azure cloud infrastructure via 3G/4G connectivity.

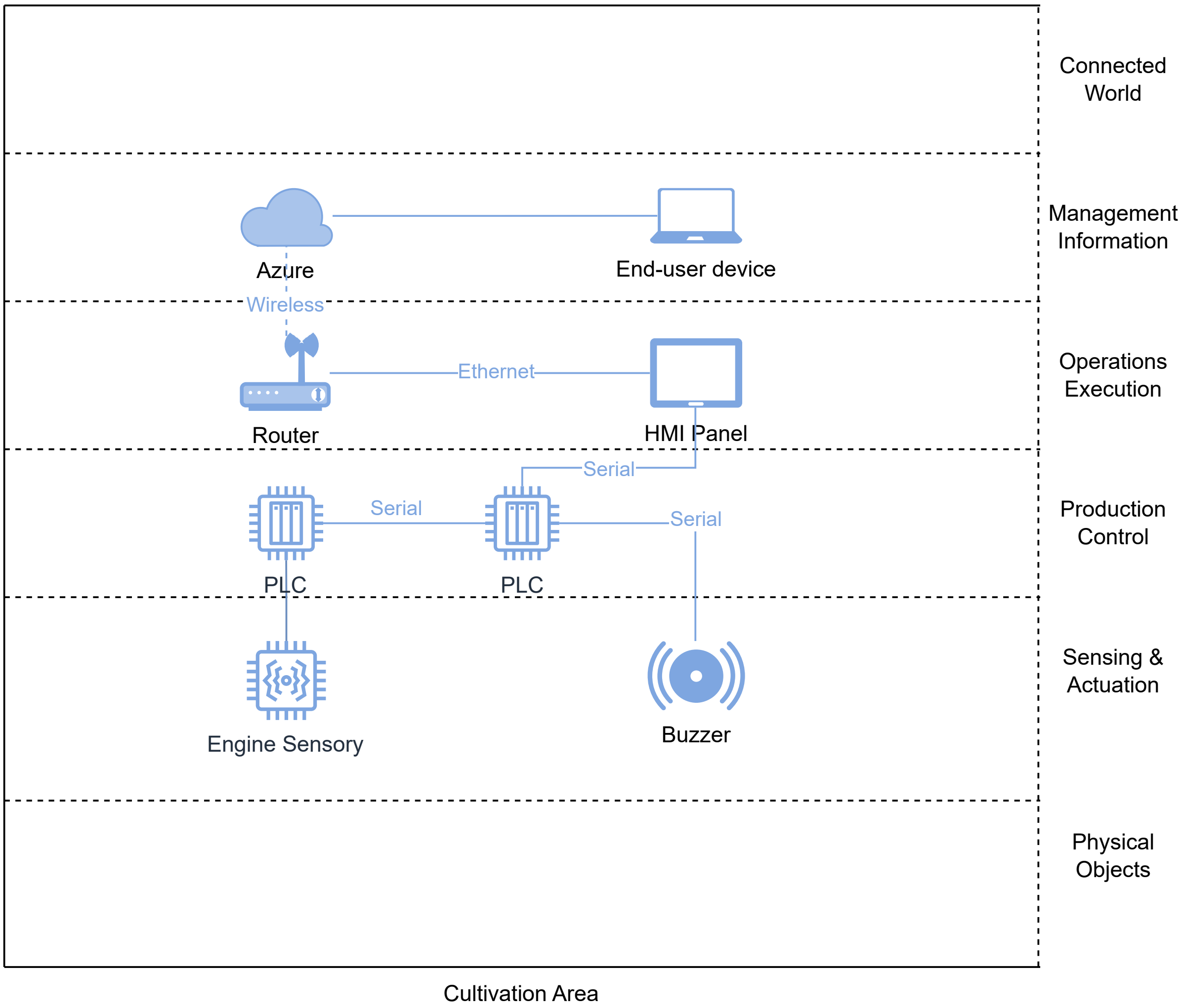

Figure 5.3: Integration layer for 'Monitor Condor Engine'

Development of the Business Layer

The business layer is designed to accommodate the business processes, services, and organizations that are connected to the use case. It also encompasses the associated business objectives, economic considerations, and regulatory restrictions. The business layer (figure 5.4) shows the area which is affected by the use cases. There are two business processes related to the use case: real-time engine monitoring and being able to access the engine data through the portal.

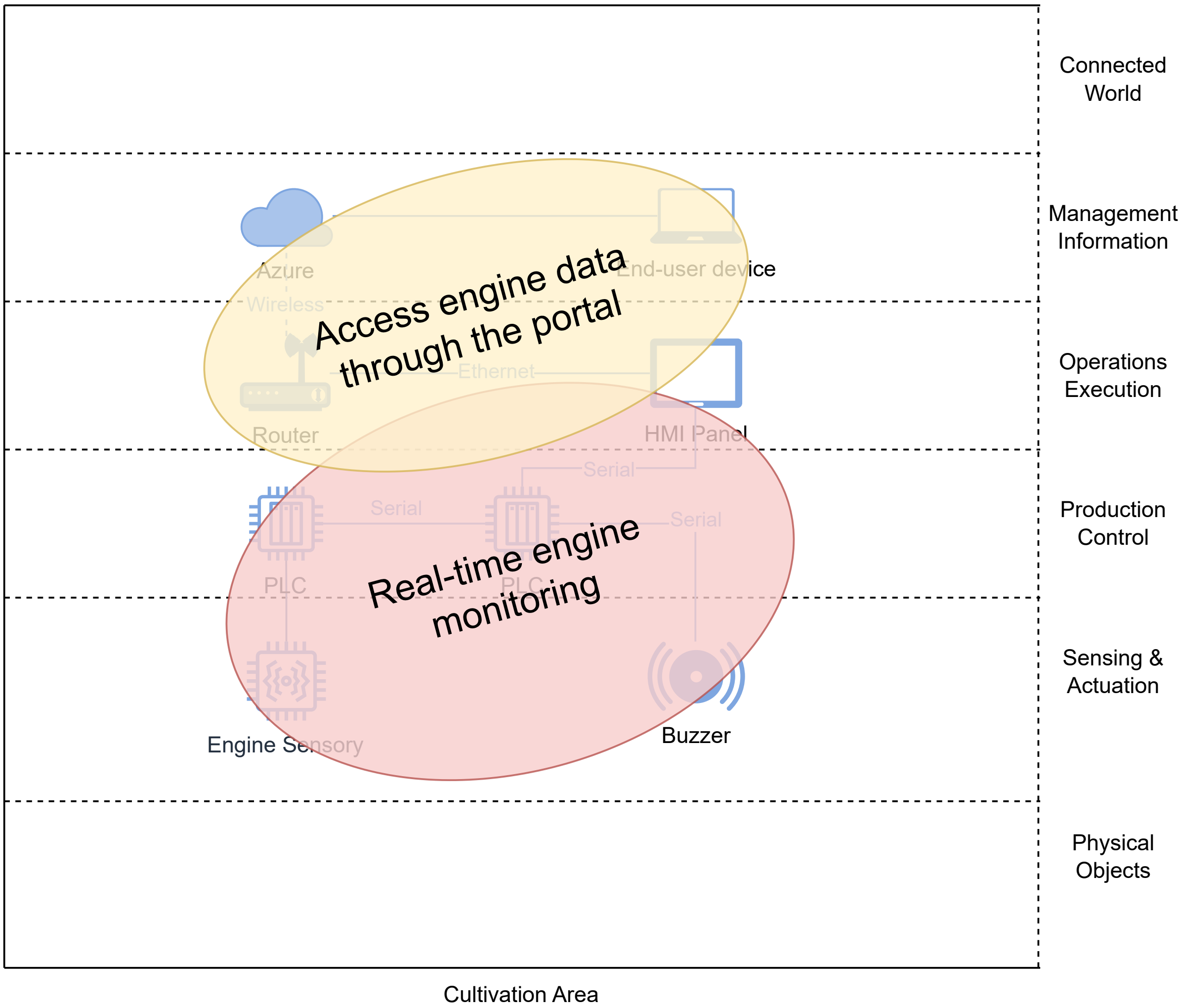

Figure 5.4: Business layer for 'Monitor Condor Engine'

Development of the Function Layer

Next, we have the function layer. It draws its features from use cases. The engine sensory monitors various engine metrics, while the buzzer audibly alerts the Condor's driver. The PLC processes engine data and controls the buzzer, also sending alerts to the HMI in case of engine issues. The HMI displays alerts and transfers sensor data to Azure. Azure stores this data and presents 'My Agrifac' to users.

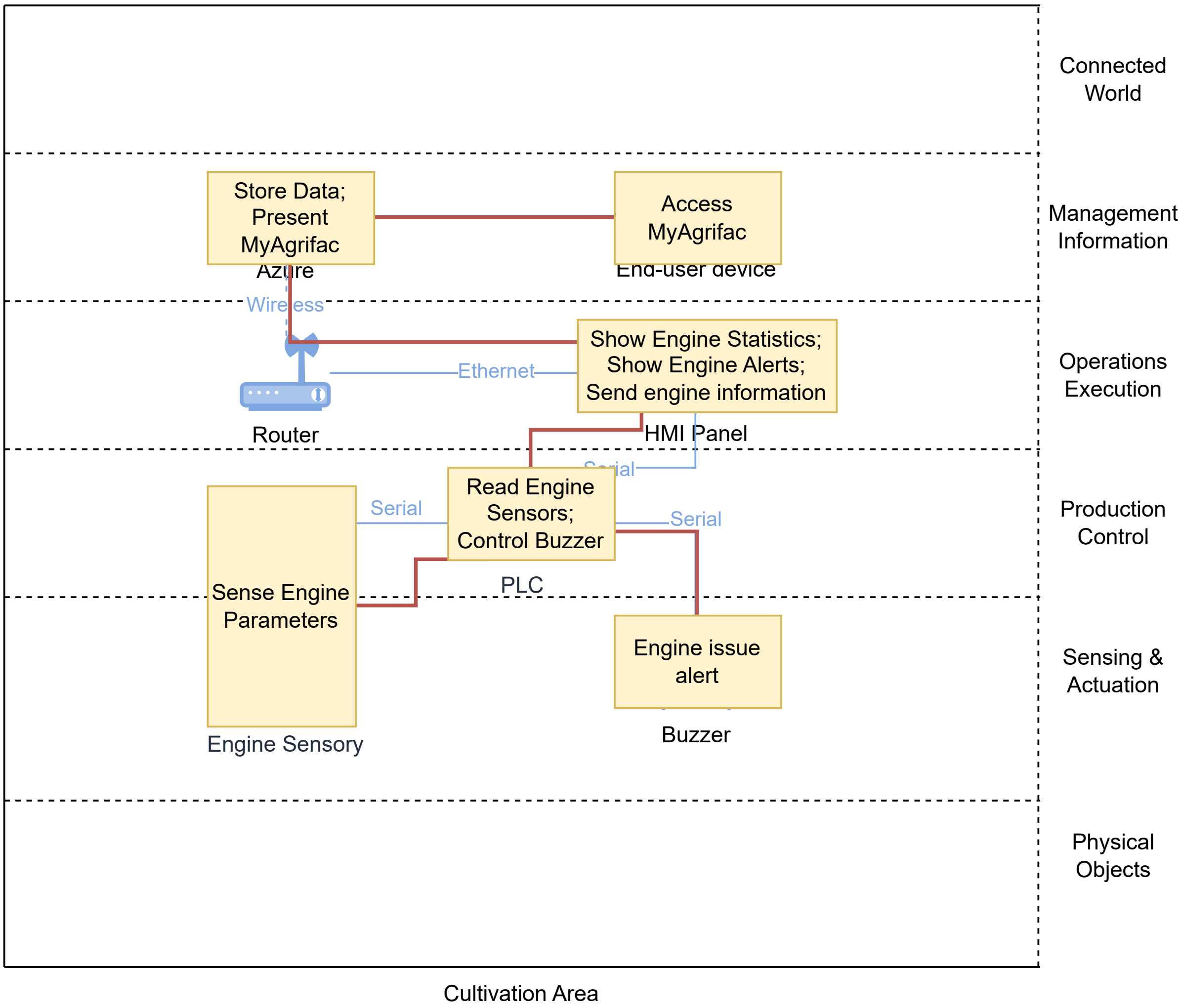

Figure 5.5: Function layer for 'Monitor Condor Engine'

Development of the Information Layer

The data layer characterizes the information utilized and shared among functions, services, and components. Figure 5.6 displays the outcome of mapping the shared information among the system's components.

The term "engine measurements" is an abstract concept that refers to the various data the engine provides. These include oil and fuel pressure, both liquid and air temperatures, and coolant level.

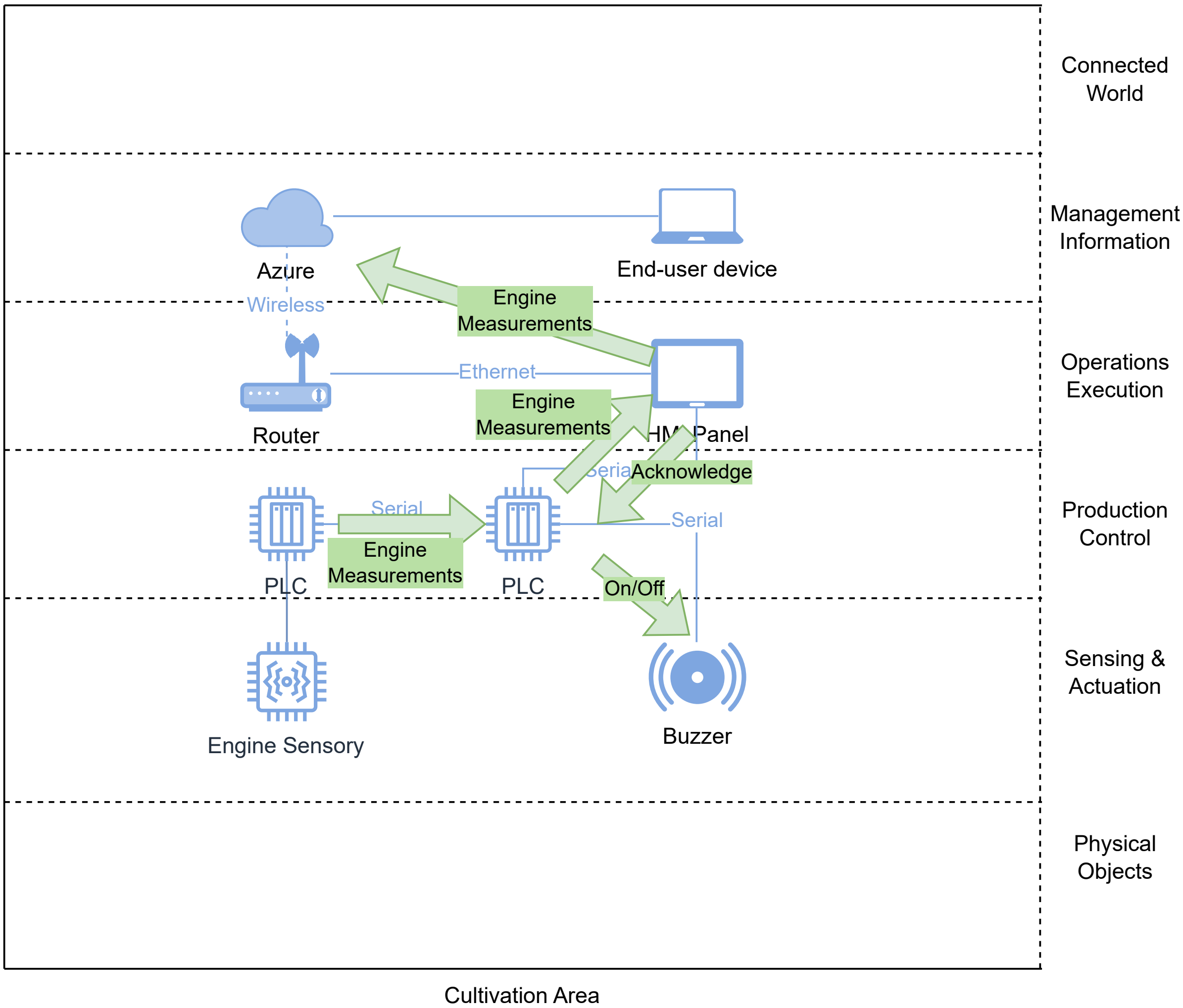

Figure 5.6: Information layer for 'Monitor Condor Engine'

Development of the Communication Layer

The communication layer details protocols and technologies that ensure the interoperable transfer of information among the system's components. Suitable protocols are determined based on the information objects, canonical data models, and the non-functional requirements of the use case. Depicted in Figure 5.7, the communication layer illustrates the protocols used for data exchange of the required information between components.

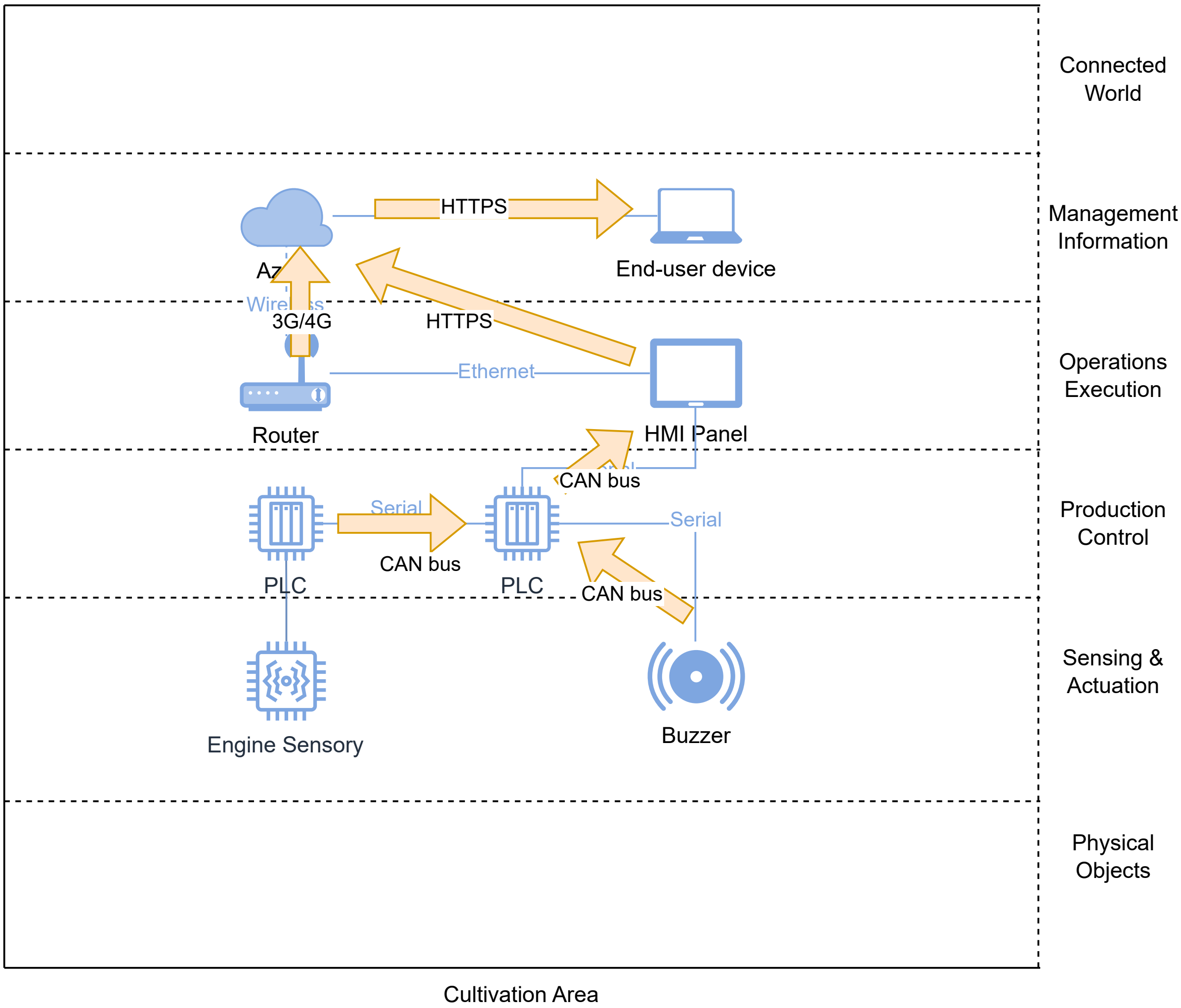

Figure 5.7: Communication layer for 'Monitor Condor Engine'

5.2 Use Case 'Precision Spot Spraying'

Precision Spot Spraying is an innovative use case of Agrifac's Condor Field Sprayer that leverages GPS technology and data-driven insights. This approach involves the creation of a prescription map, typically generated using advanced aerial data captured by drones or satellites. This map, detailing the exact needs and conditions of the field, is then uploaded to the onboard panel of the Condor sprayer.

Using GPS technology and the information from the prescription map, the Condor sprayer can treat each plant individually. This high precision ensures that the correct amount of spray is distributed to each plant, optimizing resources. This method minimizes wastage and maximizes productivity, leading to more efficient and sustainable agricultural operations.

Precision Spot Spraying, which integrates data analysis, GPS technology, and automation, offers enhanced accuracy and efficiency in plant treatment. This approach is a notable development within the field of precision agriculture.

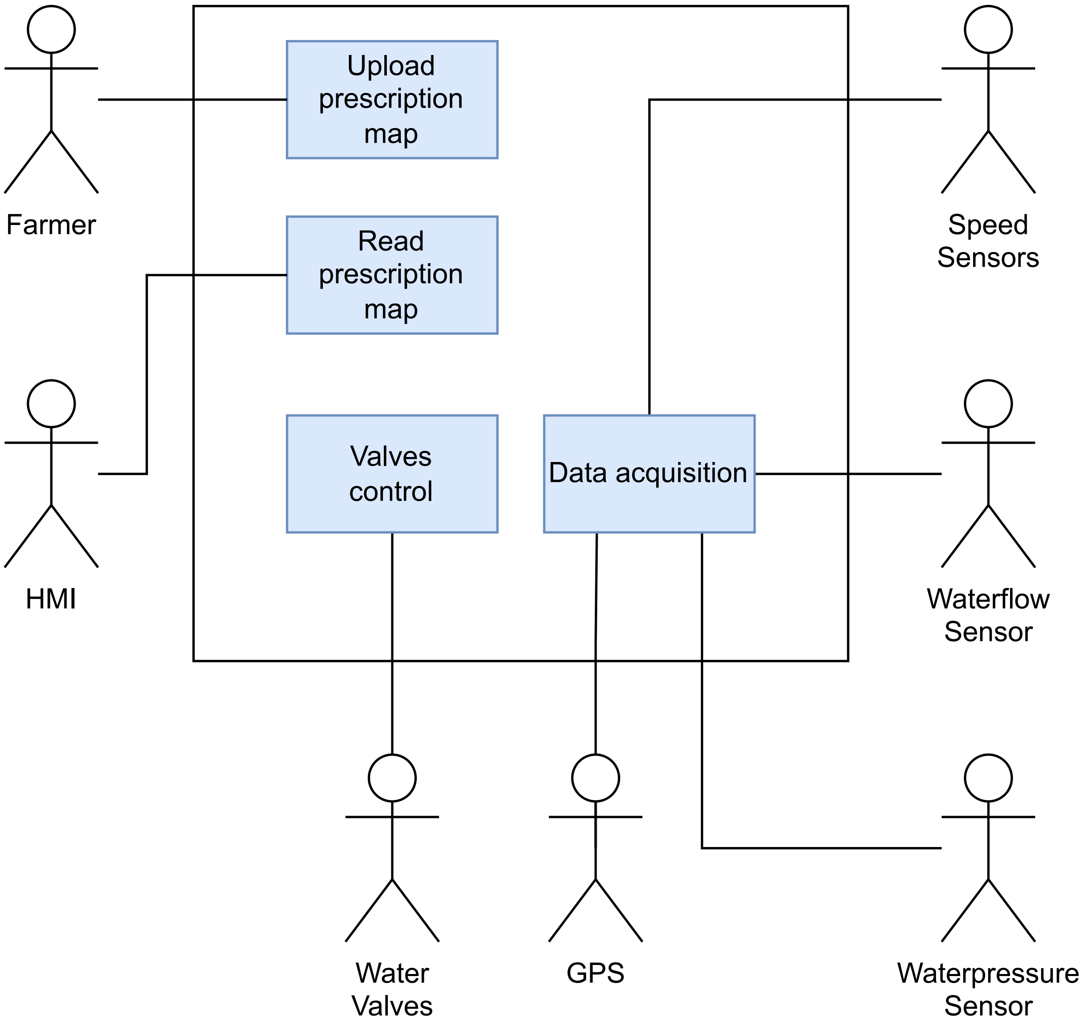

Figure 5.8: Use case diagram for 'Precision Spot Spraying'

The Agrifac Condor is designed without an integrated GPS module. This decision is rooted in the understanding that many farmers already own GPS modules. Farmers frequently prefer to swap these modules between machinery when needed because of the substantial investment cost of acquiring a GPS. This necessitates the system's versatility, ensuring interoperability with various GPS modules from multiple manufacturers, such as John Deere, Trimble, Raven, and Topcon.

Actor	Actor Type	Description	Manufacturer/Model
Farmer	End-user	The farmer is an individual who owns or works on a farm, engaging in agricultural activities such as crop cultivation, livestock rearing, and other related tasks. They drive the Condor field sprayer	-
HMI	System	The HMI is a user interface system that allows farmers to interact with the Precision Spot Spraying system, manage sensor data, and control operation.	Christ Windows 10 Embedded
Speed Sensors	Device	Speed sensors are used to monitor and control the speed of the sprayer, ensuring accurate and effective spot spraying.	-
Water Flow Sensor	Device	The water flow sensor measures the water flow in the sprayer system, allowing for precise control of the amount of spray used.	-
Water Pressure Sensor	Device	The water pressure sensor measures the water pressure in the system, which can influence the spray pattern and effectiveness.	-
Water Valves	Devices	Water valves control water flow or pesticides, allowing for precise application on specific spots as the prescription map dictates.	-
GPS	Device	The GPS calculates the current location of the device.	-

Table 5.2: List of actors 'Precision Spot Spraying'

Development of the Asset Layer

The content of the asset layer is derived from the list of actors (see table 5.2). These actors are placed in their appropriate domain and zone (5.9).

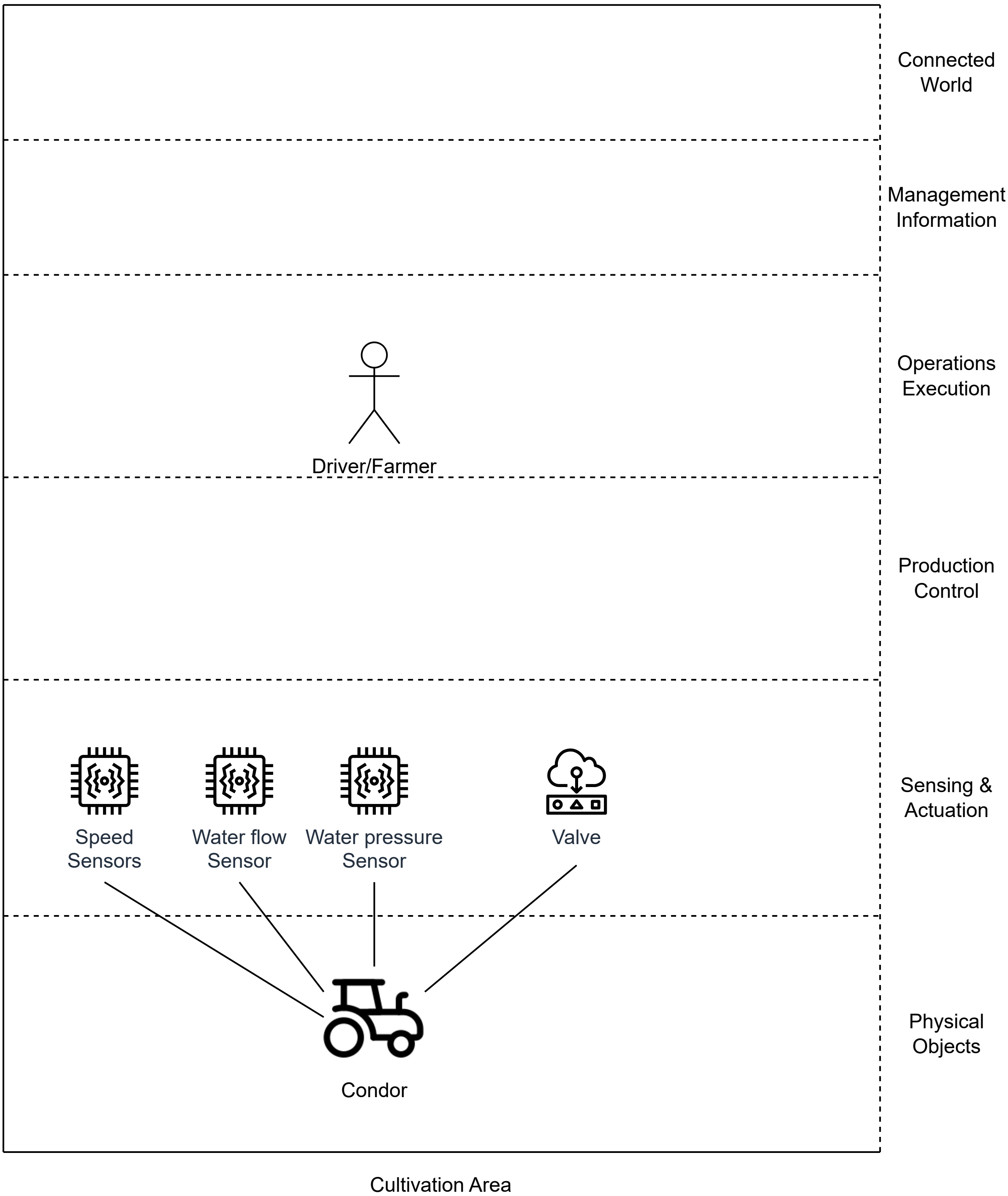

Figure 5.9: Asset layer for 'Precision Spot Spraying'

Development of the Integration Layer

After identifying the asset layer, we must bridge the physical assets with the digital world through the integration layer (5.3).

The field sprayer comprises an array of valves, each regulated by an individual microcontroller. This microcontroller interfaces with diverse sensors via a wired CAN bus, which then communicates directly with a PLC. The PLC is tethered to the HMI through a dedicated CAN bus connection. The GPS modules interface with the HMI through a serial connection. Furthermore, a USB device transfers the prescription map directly to the Condor field sprayer, thereby enhancing its operational efficiency and precision.

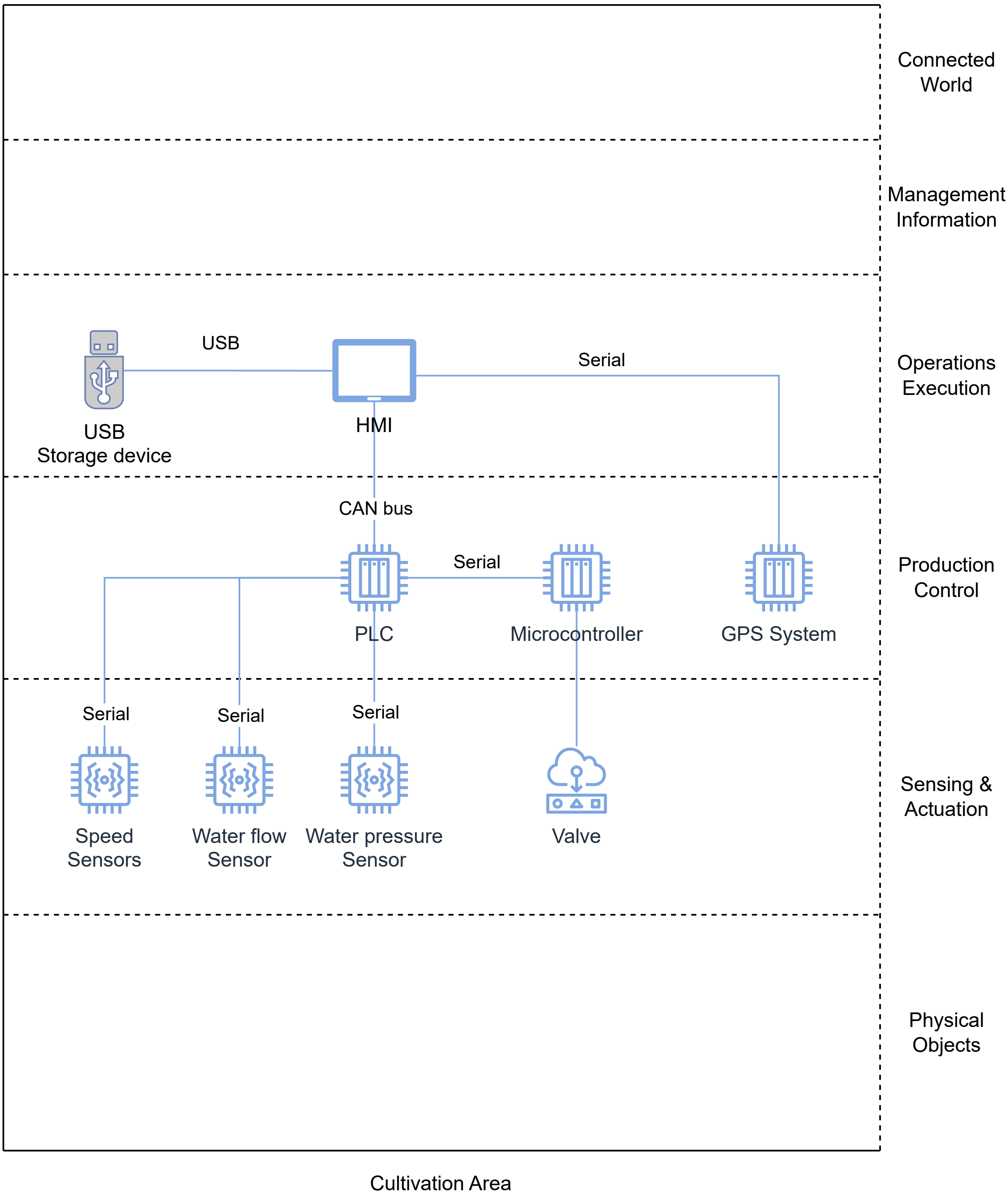

Figure 5.10: Integration layer for 'Precision Spot Spraying'

Development of the Business Layer

The business layer (figure 5.4) shows the area which is affected by the use cases. There is one business objective related to the system, high accurate spot spraying.

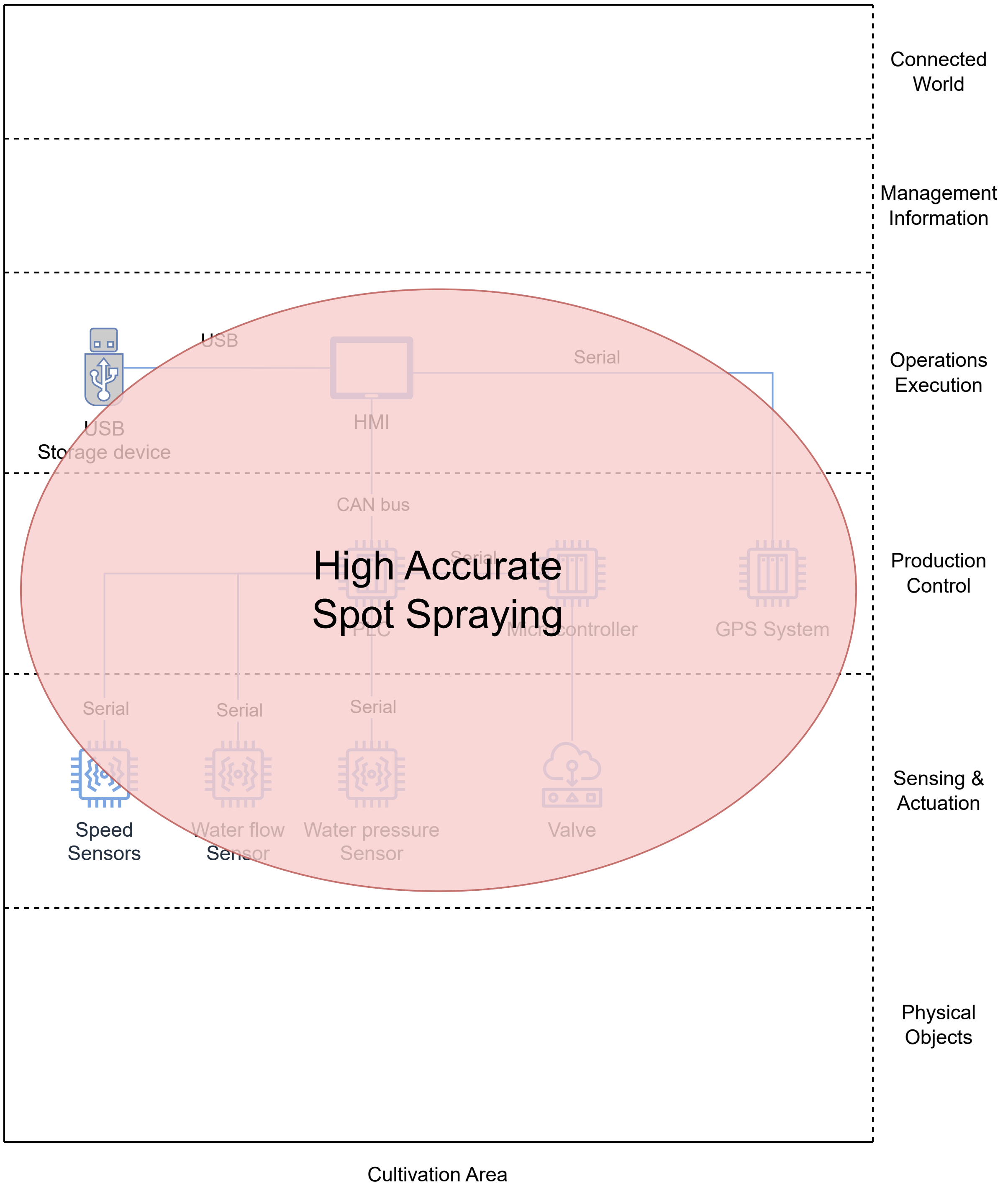

Figure 5.11: Business layer for 'Monitor Condor Engine'

Development of the Function Layer

Next, we have the function layer. It draws its features from use cases. The sensors monitor parameters such as velocity, water flow, and pressure. The HMI subsequently interprets the data from the prescription map. It uses real-time GPS coordinates to calculate the optimal spray rate for the precise location. By combining the data from the sensors with the estimated spray rate, the PLC can manage each valve individually, ensuring an efficient and accurate response for the current location.

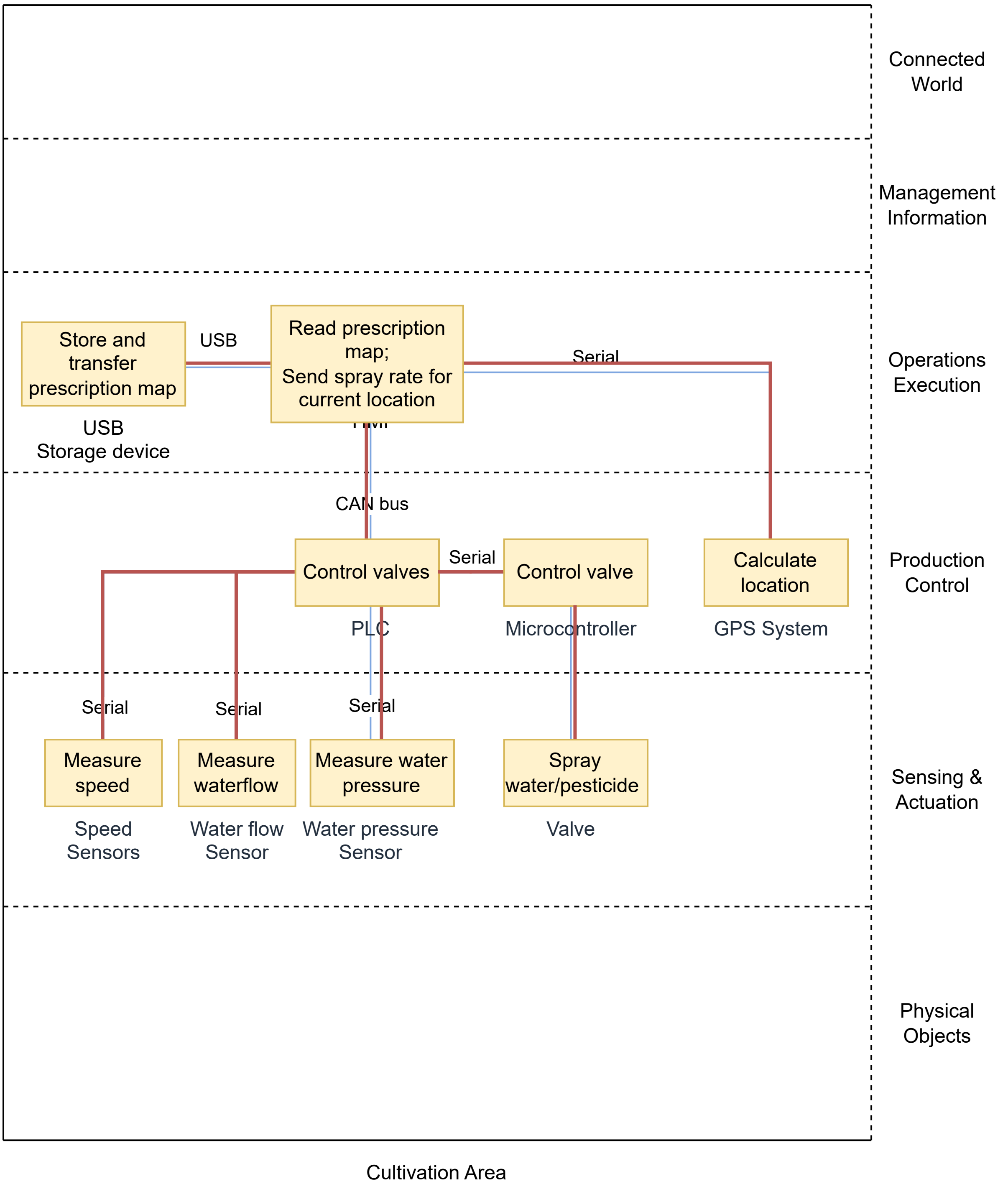

Figure 5.12: Function layer for 'Precision Spot Spraying'

Development of the Information Layer

To precisely apply water or pesticides, various data must be shared between the system's various components. The sensors dedicated to measuring speed, water flow, and water pressure convey their respective data in units of millimeters per second (mm/s), milliliters per minute (ml/min), and bar. Utilizing the geographical coordinates derived from the GPS and the prescription map, the HMI is designed to compute the optimal spray rate for the current location. This calculated spray rate is subsequently harmonized with the data gleaned from the sensors. The PLC then processes this information, transmitting each valve's frequency and duty cycle to the associated microcontroller. Upon receiving this instruction, the microcontroller, in turn, generates an electrical signal that controls the operation of the valve, dictating its opening and closing motions.

The prescription map is uploaded as a shapefile to the HMI. This file is a geospatial vector data format for geographic information system (GIS) software.

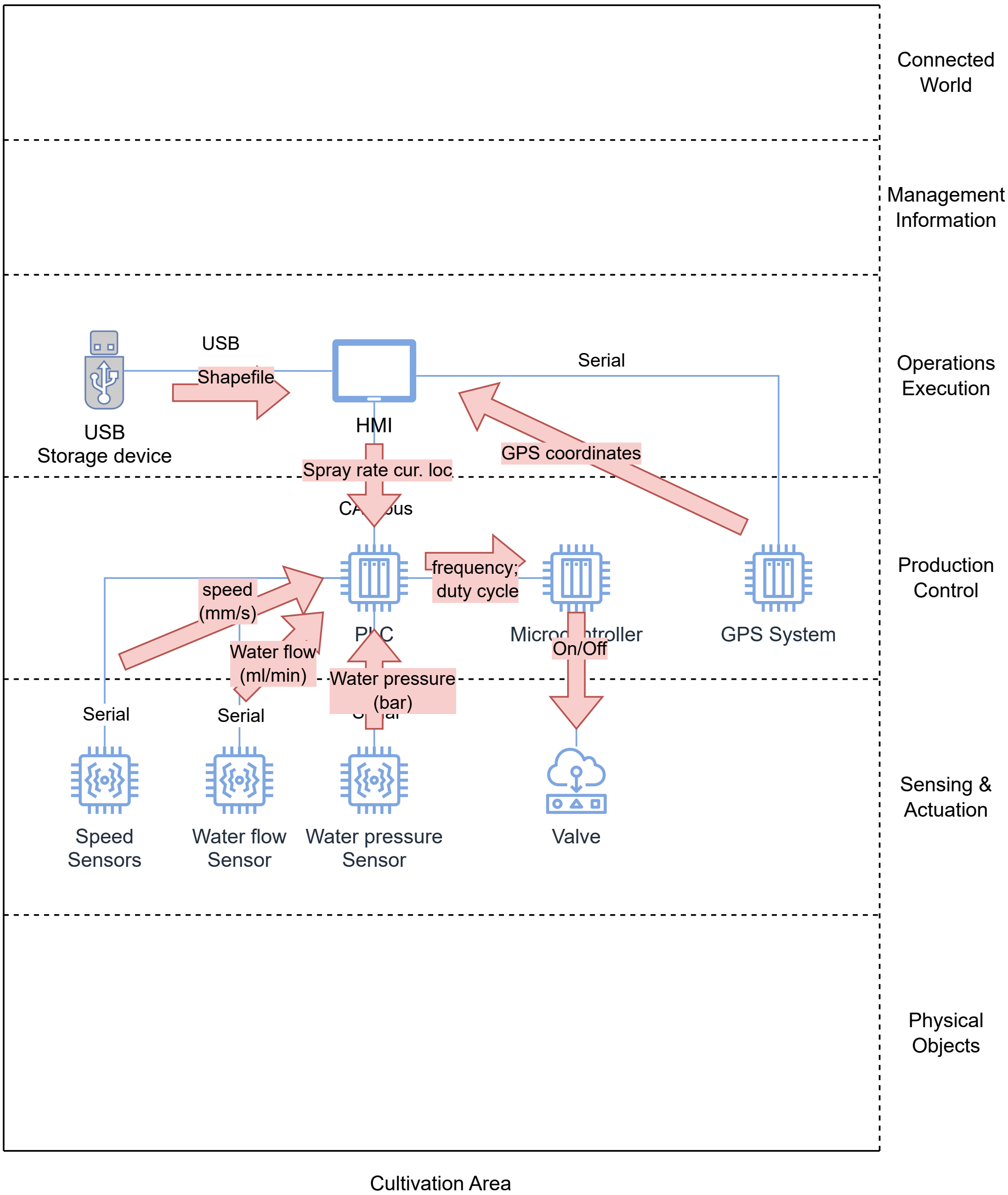

Figure 5.13: Information layer for 'Precision Spot Spraying'

Development of the Communication Layer

To facilitate the transmission of the diverse data types, a selection of communication protocols has been implemented. The sensors, HMI, and microcontrollers establish a network with the PLC via the CAN bus. The GPS Module interfaces with the computer, employing the NMEA 0183 protocol. This protocol's widespread usage among GPS manufacturers has led to its selection, guaranteeing broad compatibility and consistent performance across an array of GPS units.

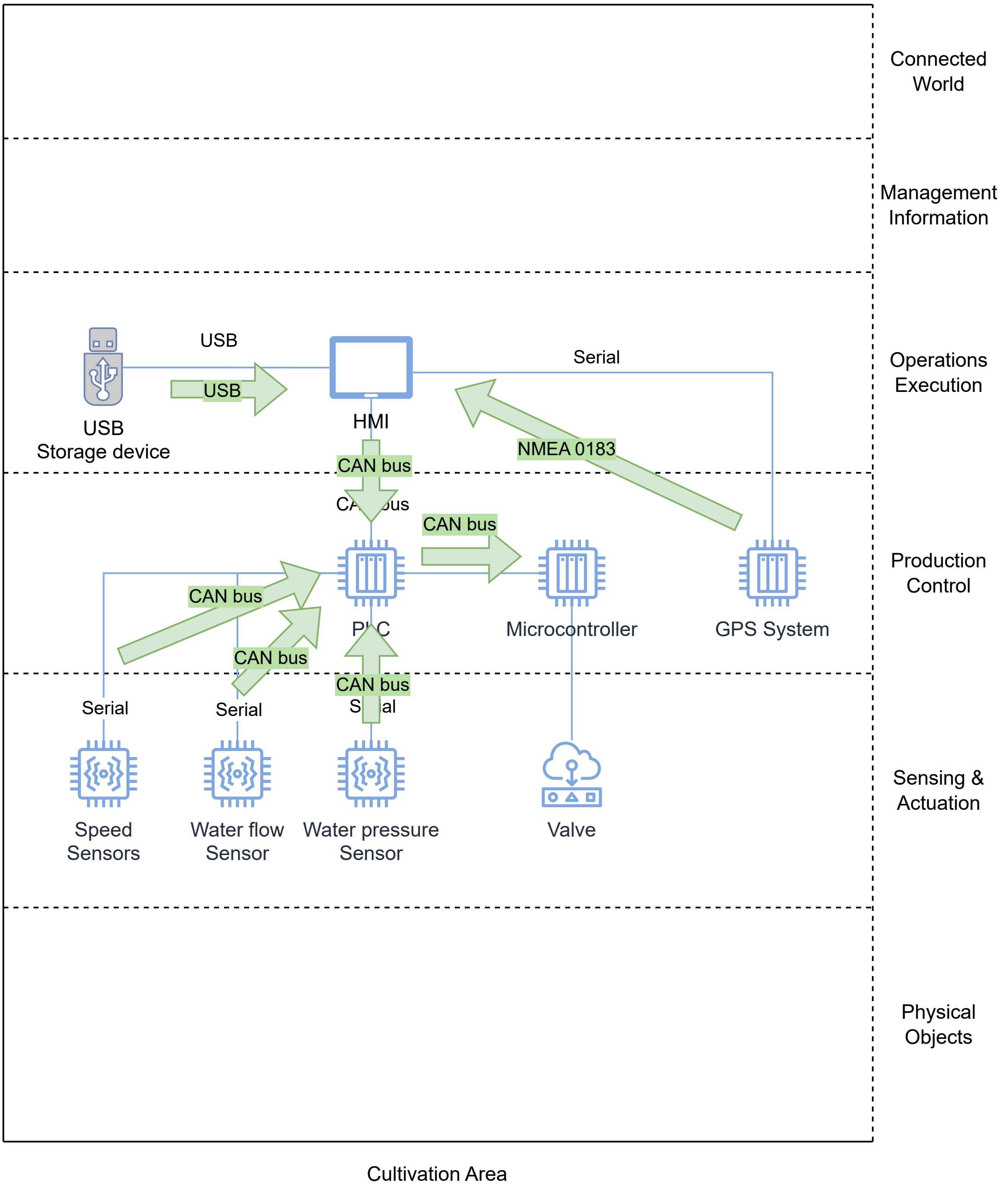

Figure 5.14: Communication layer for 'Precision Spot Spraying'

Chapter 6

Discussion

In our research, we tried to answer the question “How can an architecture framework be designed to support the development of interoperable smart farming systems?”. In chapter 4, we propose this framework, the Smart Farming Architecture Model (SFAM) Framework, acknowledging interoperable complexities inherent to modern agricultural systems. It was found that existing FMISs often tend to be specialized, focusing on one specific task on the farm, compelling farmers to use multiple FMISs to meet their diverse needs. Lacking seamless integration leading to problems with interoperability between applications and platforms. Because of this, our framework was developed.

Finding 1: Our validation demonstrated that the framework is not only effective in mapping both use cases effectively, but also versatile enough to include a broad range of elements, from tangible like machinery and personnel to digital interfaces such as Human-Machine Interfaces (HMIs), sensors, and actuators.

In collaboration with a software architect from Agrifac, we tested our framework against two of their use cases. These use cases were structured following the methodology outlined in chapter 4.3. The results were promising; our framework effectively mapped both use cases.

During validation, it was observed that the framework could successfully map a wide array of elements in smart farming systems. This includes tangible elements like machinery and field personnel, as well as digital components like HMIs, sensors, and actuators. The framework's ability to map both tangible and digital elements underscores its potential for providing a comprehensive view of complex agricultural systems.

Finding 2: The framework clearly visualizes different interoperable layers, from business elements to technical resources and physical assets.

The architecture incorporates distinct interoperable layers, which serve to represent various aspects of a system's overall interoperability. The layout is consistently uniform across each layer, enhancing readability and interpretation. This approach affords stakeholders a clear visualization of each layer, whether it be business-focused elements, technical resources, or physical assets.

Our framework draws inspiration from the SGAM framework, crafted initially for developing interoperable smart grid systems. We successfully mapped various interoperable components and data within our model in conjunction with RAMI 4.0's interoperable layers. The categorization of components by control level, derived from ISA 95 and Verdouw *et al.*'s research, proved effective for the organization of components.

Finding 3: The framework offers a broad ”system of systems” perspective. Integrating the framework with viewpoints from Verdouw *et al.* might be beneficial for a more detailed view on software implementation.

While our framework doesn't delve deeply into the software implementation of a system, it offers a holistic ”system of systems” perspective. For a more granular view of software implementation, it would be advantageous to integrate our model with the viewpoints proposed by Verdouw *et al.*

During the validation phase, we encountered some challenges. For instance, managing numerous components can become cumbersome within a static diagram. A potential solution could be creating dedicated software or a plugin akin to the SGAM toolbox to integrate with enterprise architect and UML platforms. This would facilitate more efficient design processes for smart farming systems.

6.1 Threats to Validity

Several factors may impact the validity of our findings. Primarily, our validation is constrained to mapping two existing use cases at Agrifac within the SFAM's cultivation area domain. Although Appendix A demonstrates our theoretical mapping of FMIS features across various SFAM domains, it declares that a diverse range of FMIS use cases can be structured within our framework. While this theoretically suggests adaptability across other SFAM domains, we lack empirical evidence to substantiate this claim. Future work should aim to expand our research to encompass a broader range of SFAM domains, ensuring a more comprehensive validation of our framework's versatility.

Moreover, there is potential vulnerability due to the limited breadth of expert feedback from the agritech sector. Most insights incorporated into this research are literature-derived and later corroborated by a single software architect and domain expert. This limits the range of professional perspectives, which may not entirely represent the broader agritech community.

Chapter 7

Conclusion

In response to the research question posed: "How can an architecture framework be designed to support the development of interoperable smart farming systems?", our research presents a solution in the form of the Smart Farming Architecture Model Framework. It provides a smart farming architecture framework with the addition of a methodology for applying the framework. The framework has been applied to two already existing use cases to validate the comprehensiveness of the framework. It distinctively visualizes interoperable layers spanning from business elements to technical resources and physical assets.

While drawing inspiration from established frameworks like the SGAM and benefiting from the insights of the ISA 95 model and Verdouw *et al.*'s research, the Smart Farming Architecture Model Framework emphasizes a holistic "system of systems" perspective. Yet, a more detailed view on software implementation can potentially be addressed by combining our framework with different architecture views from Verdouw *et al.*

7.1 Future Work

Challenges encountered during the validation phase, like the static nature of diagrams, suggest opportunities for further refinement, possibly through the development of specialized software or plugins. This could streamline the design processes for future smart farming systems.

However, it is essential to approach these findings with caution. Our validation, while promising, was limited in scope, focusing on two specific use cases from Agrifac. While theoretical mappings suggest the framework's adaptability across various SFAM domains, empirical evidence is lacking. Its full potential and versatility in the broader agritech sector remain to be explored in future research, especially through expansive validation and more diverse expert feedback.

Acknowledgements

I would like to extend my gratitude to Dr. Ir. M.C. Bor from the University of Amsterdam for his academic guidance and expertise. Similarly, I'd like to thank Martin van de Goor, software architect and also my project supervisor at Info Support.

I am grateful to Roelof van 't Ende and Simao Soares for giving me the opportunity to validate my research at Agrifac.

To my friends and fellow students from the Master's in Software Engineering, Tim van Ekert and Donovan Schaafsma, for the countless brainstorming sessions.

And to my parents, Peter Kok and Karin Kok, as well as my girlfriend, Anneloes de Wit, for their endless support during the writing of my thesis.

Bibliography

[1] R. Bongiovanni and J. Lowenberg-DeBoer, "Precision agriculture and sustainability," *Precision agriculture*, vol. 5, pp. 359–387, 2004.

[2] J. Stanley, G. Preetha, and J. Stanley, *Pesticide toxicity to non-target organisms*. Springer, 2016, vol. 502.

[3] N. J. Hawkins, C. Bass, A. Dixon, and P. Neve, "The evolutionary origins of pesticide resistance," *Biological Reviews*, vol. 94, no. 1, pp. 135–155, 2019.

[4] G. McIsaac, "Surface water pollution by nitrogen fertilizers," *Encyclopedia of Water Science*, vol. 950, 2003.

[5] M. Manna *et al.*, "Long-term effect of fertilizer and manure application on soil organic carbon storage, soil quality and yield sustainability under sub-humid and semi-arid tropical india," *Field crops research*, vol. 93, no. 2-3, pp. 264–280, 2005.

[6] E. Bwambale, F. K. Abagale, and G. K. Anornu, "Smart irrigation monitoring and control strategies for improving water use efficiency in precision agriculture: A review," *Agricultural Water Management*, vol. 260, p. 107 324, 2022.

[7] E. F. I. Raj, M. Appadurai, and K. Athiappan, "Precision farming in modern agriculture," in *Smart Agriculture Automation Using Advanced Technologies: Data Analytics and Machine Learning, Cloud Architecture, Automation and IoT*, Springer, 2022, pp. 61–87.

[8] J. Tummers, A. Kassahun, and B. Tekinerdogan, "Obstacles and features of farm management information systems: A systematic literature review," *Computers and electronics in agriculture*, vol. 157, pp. 189–204, 2019.

[9] C. Bahlo, P. Dahlhaus, H. Thompson, and M. Trotter, "The role of interoperable data standards in precision livestock farming in extensive livestock systems: A review," *Computers and electronics in agriculture*, vol. 156, pp. 459–466, 2019.

[10] S. Neethirajan, "Recent advances in wearable sensors for animal health management," *Sensing and Bio-Sensing Research*, vol. 12, pp. 15–29, 2017.

[11] J. Fulton, E. Hawkins, R. Taylor, and A. Franzen, "Yield monitoring and mapping," *Precision agriculture basics*, pp. 63–77, 2018.

[12] P. Clements, D. Garlan, R. Little, R. Nord, and J. Stafford, "Documenting software architectures: Views and beyond," in *25th International Conference on Software Engineering, 2003. Proceedings.*, IEEE, 2003, pp. 740–741.

[13] ISO/IEC/IEEE, *ISO/IEC/IEEE 42010:2022 Software, systems and enterprise — Architecture description*, `https://www.iso.org/standard/74393.html`, 2022.

[14] I. F. Team, "Interoperability context-setting framework," 2007.

[15] M. Uslar *et al.*, "Applying the smart grid architecture model for designing and validating system-of-systems in the power and energy domain: A european perspective," *Energies*, vol. 12, no. 2, p. 258, 2019.

[16] S. G. C. Group, "Smart grid reference architecture," CEN-CENELEC-ETSI, Technical Report, Nov. 2012. [Online]. Available: `https://energy.ec.europa.eu/publications/smart-grid-reference-architecture_en`.

[17] M. Gottschalk, M. Uslar, and C. Delfs, *The use case and smart grid architecture model approach: the IEC 62559-2 use case template and the SGAM applied in various domains*. Springer, 2017.

[18] A. Bousdekis and G. Mentzas, "Enterprise integration and interoperability for big data-driven processes in the frame of industry 4.0," *Frontiers in Big Data*, vol. 4, 2021, ISSN: 2624-909X. DOI: 10.3389/fdata.2021.644651. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fdata.2021.644651.

[19] C. Verdouw, H. Sundmaeker, B. Tekinerdogan, D. Conzon, and T. Montanaro, "Architecture framework of iot-based food and farm systems: A multiple case study," *Computers and Electronics in Agriculture*, vol. 165, p. 104 939, 2019.

[20] *Mapping to the isa-95 model - industrial iot architecture patterns*, en-US. [Online]. Available: https://docs.aws.amazon.com/whitepapers/latest/industrial-iot-architecture-patterns/mapping-to-the-isa-95-model.html.

[21] C. Koulamas and A. Kalogeras, "Cyber-physical systems and digital twins in the industrial internet of things [cyber-physical systems]," *Computer*, vol. 51, no. 11, pp. 95–98, 2018.

[22] P. Melo and E. Godoy, "Controller interface for industry 4.0 based on rami 4.0 and opc ua," Jun. 2019, pp. 229–234. DOI: 10.1109/METROI4.2019.8792837.

[23] C. Sørensen *et al.*, "Conceptual model of a future farm management information system," *Computers and Electronics in Agriculture*, vol. 72, no. 1, pp. 37–47, 2010, ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2010.02.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168169910000396.

[24] R. Tomasi *et al.*, "The iof2020 use case architectures and overview of the related iot systems," IoF2020 WP3, Tech. Rep., 2018, p. 221. [Online]. Available: https://www.iof2020.eu/deliverables/d3.2-uc-architectures-v2-final.pdf.

Appendix A

Mapping of FMIS Features to SFAM Domains

The following table provides a comprehensive mapping of 81 FMIS features to the SFAM domains, based on Tummers *et al.*'s study on features and obstacles of FMIS [8]. The data is sorted by the frequency of occurrences in the materials studied by Tummers *et al.* The aim of this table is to validate the comprehensiveness of the SFAM domains by aligning them with the specific features identified in the aforementioned study. Each feature has been assigned to the corresponding SFAM domain, indicated by an 'x' in the appropriate column. The resulting matrix offers an in-depth overview of the distribution of the FMIS features across the SFAM domains.

FMIS Features	Cultivation Area	Livestock	Storage	Processing	Distribution	Occurences
Financial management	x	x	x	x	x	17
Reporting	x	x	x	x	x	14
Data acquisition	x	x	x	x	x	13
Operation plan generation	x	x	x	x	x	10
Crop management	x					10
Resource management	x	x	x	x	x	9
Equipment management	x					9
Field monitoring	x					9
Data processing	x	x	x	x	x	9
Fertilization management	x					9
Human resource management	x	x	x	x	x	7
Weather service	x					7
Data management	x	x	x	x	x	7
Field management	x					7
Accounting	x	x	x	x	x	7
Inventory management	x	x	x	x	x	7
Decision support	x	x	x	x	x	6
Operation management	x	x	x	x	x	6
Yield estimation	x					6
Field mapping	x					6
GIS management	x	x				6
Irrigation management	x					6
Sensor management	x	x	x	x		6
Traceability	x	x	x	x	x	6
Expert knowledge	x	x	x	x	x	5
Livestock management	x	x	x	x	x	5

Continues on next page

Table A.1 – *Continued from previous page*

FMIS Features	Cultivation Area	Livestock	Storage	Processing	Distribution	Occurences
Sales					x	5
Data transfer	x	x	x	x	x	5
Data storage	x	x	x	x	x	5
Disease management	x	x				5
Harvest management	x					4
Machinery tracking	x	x		x	x	4
Pesticide management	x					4
Scheduling	x	x	x	x	x	4
Work management	x	x	x	x	x	4
Knowledge management	x	x	x	x	x	4
Legal management	x	x	x	x	x	4
Activity monitoring		x	x	x	x	4
Customer management					x	4
Alerting	x	x	x	x	x	4
Production monitoring	x	x				3
Seed management	x					3
Yield monitoring	x					3
Parameter monitoring	x	x	x	x	x	3
Transport management					x	2
Calibration management	x					2
Experience management	x	x	x	x	x	2
Marketing and sales					x	2
Reproductivity management	x					2
Weighing management		x				2
Data sharing	x	x	x	x	x	2
Driver assistance	x					2
Energy management	x	x	x	x	x	2
Health management		x				2
Information search	x	x	x	x	x	2
Model production parameters	x	x		x		2
Performance management	x	x	x	x	x	2
Scenario simulation	x	x	x	x	x	2
Strategic planning	x	x	x	x	x	2
Technology management	x	x	x	x	x	2
Best practice	x	x	x	x	x	1
Collect produce information	x			x		1
Communication	x	x	x	x	x	1
Condition management	x	x				1
Delivery management					x	1
Feed management					x	1
Grazing management					x	1
Herd management					x	1
Printing				x	x	1
Real estate management	x	x	x	x	x	1
Remote controlling	x	x	x	x	x	1
Risk analysis	x	x	x	x	x	1
Society management	x	x	x	x	x	1

Continues on next page

Table A.1 – *Continued from previous page*

FMIS Features	Cultivation Area	Livestock	Storage	Processing	Distribution	Occurences
Supply management	x	x	x	x	x	1
Task file management	x	x	x	x	x	1
Task supervision	x	x	x	x	x	1
Vision planning	x	x	x	x	x	1
B2B Collaboration	x	x	x	x	x	1
Company information	x	x	x	x	x	1
Environmental monitoring	x	x				1
Planting management	x					1

Table A.1: Features across the FMAS domains

Glossary

Architecture modeling framework A structured approach for representing, analyzing, and designing architectural frameworks and models to address specific requirements or challenges.. 5

Data-driven techniques Methods and approaches that rely on the analysis and interpretation of data to guide decision-making and optimize processes.. 4

Fertilizer application The process of applying fertilizers to crops or soil to provide essential nutrients for plant growth and productivity.. 4

Internet of Things (IoT) A network of interconnected devices embedded with sensors, software, and other technologies to enable communication and data exchange.. 4

Livestock welfare The well-being and ethical treatment of animals raised for agricultural purposes, including factors such as health, housing, and handling.. 4

Pest infestations The presence and proliferation of pests, such as insects, weeds, or diseases, that can cause damage to crops or livestock.. 4

Remote sensing The acquisition of information about an object or phenomenon from a distance, typically using sensors mounted on aircraft or satellites.. 4

Acronyms

DER Distributed Energy Resource. 9
EMAM Electric Mobility Architecture Model. 10
FMIS Farm Management Information Systems. 4, 11, 13, 17, 43–45
ICT Information and Communication Technology. 10, 11
MAF Maritime Architecture Framework. 11
RAMI 4.0 Reference Architecture Model for Industrie 4.0. 11
SCIAM Smart City Infrastructure Architecture Model. 10, 11
SFAM Smart Farming Architecture Model. 11, 13, 17, 19, 20, 43
SGAM Smart Grid Architecture Model. 9–11

List of Figures

List of Tables