Institute for Computer Science
Knowledge-Based Systems

Spatio-temporal Analysis for Semantic Monitoring of Agricultural Logistics

Dissertation Thesis

zur Erlangung des Doktorgrades (Dr. rer. nat.)
des Fachbereichs Mathematik/Informatik
der Universität Osnabrück

vorgelegt von
Henning Deeken

August 2021

First supervisor:	Prof. Dr. Joachim Hertzberg
Second supervisor:	Prof. Dr. Arno Ruckelshausen
Third supervisor:	Prof. Dr. Hans-Werner Guesgen

Someone has whipped the carpet from beneath my feet,
Someone upturned the furniture in my mind,
But oh, how rich the soil,
How wondrous the upheaval,
It's time to embark.

To dissect is to broaden the adventure,
And enrich one's tenure.
So do not blunt the surgeon's knife.

– Rou Reynolds, *Enter Shikari*

Danksagung

Bei Prof. Dr. Joachim Hertzberg möchte ich mich ganz herzlich für seine Bereitschaft, diese Arbeit zu betreuen, bedanken sowie für die offene Beratung und motivierenden Worte, wenn mir nicht klar war, ob und wie es mit dieser Arbeit weitergehen kann. So ging es dann immer weiter.

Prof. Dr. Arno Ruckelshausen danke ich dafür, dass er sich bereit erklärt hat, diese Arbeit zu begutachten. Prof. Dr. Hans-Werner Guesgen möchte ich für die Begutachtung, sowie die fachlichen Diskussionen während meines Aufenthalts an der Massey University danken.

Thomas Wiemann möchte ich für die langjährige gemeinsame wissenschaftliche Arbeit danken. Dank dir, Thomas, für all die konstruktive Kritik an Konzepten und Texten und die frühe Anleitung zum Publizieren. Beides hat diese Arbeit erst möglich gemacht.

Thilo Steckel und Florian Krampe möchte ich für die gute Zusammenarbeit im Projekt SOFiA danken. Thilo, Danke für deine Begeisterung an der akademischen Arbeit im industriellen Kontext und als Vorbild, dass sich Promotion und CLAAS nicht kategorisch ausschließen. Florian, Danke für alle Beiträge rund um die publizierten und nicht publizierten Aspekte meiner Arbeit. Ohne dich hätte diese Arbeit nicht die Form, die sie heute hat!

Ein großer Dank geht ebenfalls an die Kollegen vom Fraunhofer IML und Diebold Nixdorf aus dem SOFiA Konsortium. Insbesondere an Markus Witthaut, Dominik Sparer, Björn Künsting und Philipp Sprenger für die gute Zusammenarbeit an den gemeinsamen Publikationen.

In diesem Rahmen gilt mein Dank auch Prof. Dr. Nils Aschenbruck und Prof. Dr. Clemens Westerkamp für die fachlichen Diskussionen rund um das SOFiA Projekt.

Ebenfalls gilt mein Dank allen Kollegen der CLAAS E-Systems. Insbesondere Boris Kettelhoit, für seine gute Führung in einem spannenden und fordernden Arbeitsumfeld, durch die ich stets wachsen kann. Ein Dank der gesamten Advanced Engineering Abteilung für das angenehme Arbeitsklima. Besonders natürlich der CDMA Truppe – Ivan, Florian, Heinrich und Mohamed – für die enge und freundschaftliche Zusammenarbeit.

Auch möchte ich Theo Broeker und Sebastian Seidel für die gute Zusammenarbeit danken. Ebenfalls gilt mein Dank all den Studenten, die ich im Rahmen der letzten Jahre betreuen durfte: Philipp Behrens, Nils Niemann, Verena Kirstein, Niels Pressel, Katja Wissmann, Benjamin Karic & Lennart Niecksch.

Auch gilt mein Dank allen Freunden in Kempen, Münster, Osnabrück und andernorts.

Ein großes Dankeschön an Kai für die langjährige Freundschaft sowie Speis und Trank. Ein Dank an Basil Wahn für lange Diskussionen über wissenschaftliches Arbeiten und Science-Fiction, sowie unzählige schöne Privatkonzerte. Ebenso ein Dankeschön an alle Genussgenossen, die Herzberg Crew – Achtung Verwechslungsgefahr: das Festival, nicht der Professor – sowie die Jungs von Drive Party und Enter Shakira.

Ebenfalls möchte ich mich bei Familie Huesmann danken. Danke dafür, dass Ihr mich mit offenen Armen aufgenommen habt und die schöne Zeit mit euch.

Der größte Dank gilt meiner Familie, die ich mich immer uneingeschränkt unterstützt. Danke, Mama, Danke, Papa, für all euren Rückhalt in guten und schlechten Zeiten! Und für all das, was diesen meinen Bildungsweg möglich gemacht hat. Danke auch dir, Hauke, für die Korrekturen an dieser Arbeit, für die fachlichen Diskussionen und das Der-beste-Bruder-Sein ganz generell!

Der allergrößte Dank jedoch gilt Mira, die ich mich immer unterstützt und mir auch zeigt, wo die Grenze zwischen Arbeiten und Leben liegt. Durch dich habe ich in den letzten Jahren gelernt, wie das mit der Balance im Leben besser funktionieren kann. Es tut mir leid, dass das in den letzten Monaten jedoch nicht gut geklappt hat und unsere gemeinsame Freizeit oft unter Arbeit und Doktorarbeit gelitten hat. Damit ist jetzt Schluss.

Mira, Danke für deine Liebe und all die wunderschöne Zeit, die wir trotz Doktorarbeit hatten. Ich freue mich auf die gemeinsame Zeit ohne und alles, was für uns noch Schönes kommen mag. Ich liebe dich!

Abstract

Managing agricultural processes with significant logistics sub-processes is a challenge because coordinating a distributed fleet in a dynamic environment is difficult without proper oversight in terms of qualitative and quantitative process information. Digital assistance systems are thought to aid agricultural practitioners by providing process-related information and thus support operational decision-making or even control the logistic flow (semi-)automatically. However, their development is currently stifled by a lack of monitoring capabilities during process execution.

This thesis concerns the topic of online process monitoring for ongoing agricultural logistic processes. It discusses how to extract process knowledge from the telemetry of agricultural machines by applying spatio-semantic reasoning techniques. Our method combines spatial analysis for identifying spatial relationships between machines and their environment with semantic inference to derive formal process knowledge through ontological and rule-based reasoning.

To test our concepts, we implemented a domain-agnostic semantic mapping framework and applied it in the context of forage maize harvesting. We present custom-made ontological models and rules to represent agricultural environments and to reason about machine actors and their process states. Based on our prototype, we demonstrate how to implement automated process and service tracking in near-real-time. Finally, we discuss the role of online process analytics systems in the context of other agricultural assistance systems for farm and fleet management.

Zusammenfassung

Das Steuern von landwirtschaftlichen Prozessen mit ausgeprägten logistischen Teilprozessen ist eine Herausforderung, da die Koordinierung eines verteilten Fuhrparks in einem dynamischen Umfeld ohne einen angemessenen Überblick über qualitative und quantitative Prozessinformationen schwierig ist. Digitale Assistenzsysteme sollen den landwirtschaftlichen Praktikern durch die Bereitstellung prozessbezogener Informationen helfen und so die operative Entscheidungsfindung unterstützen oder sogar den logistischen Fluss (halb)automatisch steuern. Allerdings wird die Entwicklung dieser Systeme derzeit durch fehlende Möglichkeiten der Prozessüberwachung während der Prozessdurchführung gehemmt.

Diese Arbeit befasst sich mit der Prozessüberwachung für laufende agrarlogistische Prozesse. Es wird diskutiert, wie Prozesswissen aus der Telemetrie landwirtschaftlicher Maschinen durch die Anwendung räumlich-semantischer Schlußfolgerungstechniken extrahiert werden kann. Unsere Methode kombiniert räumliche Analyse zur Identifizierung räumlicher Beziehungen zwischen Maschinen und ihrer Umgebung mit semantischer Inferenz zur Ableitung von formalem Prozesswissen durch ontologische und regelbasierter Inferenz.

Um unsere Konzepte zu testen, haben wir ein domänenunabhängiges Framework zur Repräsentation semantischer Karten implementiert und es im Kontext der Futtermaisernte angewendet. Hierfür stellen wir maßgeschneiderte ontologische Modelle und Regeln vor, um landwirtschaftliche Umgebungen zu repräsentieren und um Schlussfolgerungen über Maschinenakteure und deren Prozesszustände zu ziehen. Auf der Grundlage unseres Prototyps demonstrieren wir, wie eine automatische Prozess- und Leistungsverfolgung in Nahe-Echtzeit umgesetzt werden kann. Abschließend diskutieren wir die Rolle von Online-Prozessanalysesystemen im Zusammenhang mit anderen landwirtschaftlichen Assistenzsystemen für das Betriebs- und Flottenmanagement.

Contents

Part I

Research Summary

Chapter 1

Introduction

In times of an increasing world population, shrinking natural resources, and climate change, agriculture must become more efficient and ecological. According to the Food and Agriculture Organization of the United Nations (FAO), an increase of agricultural production by about 60 % from 2009 and 2050 will be necessary to support our earth's population and its demand for food [10]. Hence advancements must be made along the entire agricultural production chain. This includes improvements in plant cultivation, livestock management, usage of agricultural machines, as well as food production techniques and distribution methods.

Improvements towards the usage of agricultural machinery can generally be approached in two ways. One can improve the utility of the machines by making them functionally more capable, increasing their maximal work capacity, or improving their efficiency regarding resource consumption. Alternatively or additionally, one can utilize the machines better by working with them more effectively, for example, by using them more often or decreasing downtimes during operations. Regarding better utilization, some potential lies in optimizing the way multiple machines are working together in cooperative agricultural processes. This holds especially true for harvesting processes since research has shown that the full potential of harvesting machines is usually not adequately utilized, often due to insufficiencies in the logistic chain from field to farm [201]. Therefore, improving the logistics aspects of a harvest operation promises to make the entire harvesting process more efficient.

A critical problem regarding the organization and optimization of agricultural logistic chains is the lack of proper insight into the process during its execution. This is partly due to the dynamic and distributed nature of harvesting processes. Multiple machines are dispersed across a vast and highly dynamic environment prone to technical and natural disturbances. This makes it hard for human operators to quickly oversee the current state of the process, determine its progress and manage the logistics accordingly. Digital assistance systems are thought to be an adequate remedy for this, as they can help to prepare, oversee and execute harvesting processes.

In the past two decades, digital assistance systems have become increasingly important in the agricultural sector. With affordable computing power and access to Global Navigation Satellite Systems (GNSSs), it became feasible to deploy recording of geo-referenced telemetry data onboard of agricultural machines. Furthermore, with increasing coverage with mobile broadband communication, it became industry practice to collect this data and build products for agricultural data management and analytics. This lead to the introduction of digital assistance systems into many agricultural processes. Especially, precision farming techniques have become essential factors in agriculture production and are commonly found in single machine operations, such as seeding and spraying [163]. To digitally support agricultural processes that involve multiple, cooperating machines, however, is still an open topic, and digitally support agricultural logistics is currently a goal of the agricultural community [30, 196].

In recent years, approaches to digitally (pre-)plan, schedule, and mathematically optimize harvesting operations have been studied extensively and with promising results. However, improving agricultural work during its execution requires applying planning and optimization capabilities in control systems that dynamically adapt to changing conditions and unforeseen events. To continuously monitor the process that is to be optimized while it is in execution is, therefore, a mandatory pre-requisite to controlling it optimally. However, the current state of an ongoing process is not immediately apparent from a machine's telemetry alone. It only informs about a machine's position and internal states. Usually, neither its current activity nor its relations to partner machines are explicit in the recorded data. Hence the current process state is also not explicitly available for comparison against a set of reference states. This situation currently hinders matching a previously made plan against the machine's actual work and stifles planning and control applications. It also leads to a demand for novel systems that monitor the telemetry and derives process information from it. Depending on the type of process to be monitored, such a system may require additional data or related background knowledge during this data analysis and information grounding process.

A system that performs *online process analytics* is, therefore, an important component in the context of digital assistance for agricultural logistic processes. Yet, while systems for planning and optimizing agricultural logistical processes pre-operations have been studied to some extent and offline monitoring systems, in form of the automated, post-operational documentation of agricultural work are commonly known, the topic of monitoring agricultural processes during operations and in terms of formal semantics and has been neglected in the scientific literature.

1.1 Thesis Topics

This thesis contributes to the subject of digital agriculture, as it addresses the topic of monitoring ongoing agricultural logistic processes. Here it is concerned with solving the following problem.

Problem Statement - Digital Agriculture

Modern agricultural machines participating in an agricultural work process, e.g., harvesting, provide a continuous stream of telemetry data but no explicit information about what they are doing in terms of logistical activities. From a practitioner's perspective, this limits human insight into the ongoing process and complicates manual control over the logistic process, especially when having a remote view, e.g., when being the manager of a large-scale harvesting operation. From a technical perspective, this also hinders the development of digital assistance systems for agricultural logistics since no grounding function derives explicit machine-readable information about the process, which can be used as input for planning and control systems.

However, knowledge about the process is implicit within the telemetry and can be derived by analyzing it in conjunction with additional data and background knowledge. When working towards digital assistance systems capable of planning and control, it is advisable to design such grounding functions that derive formal knowledge about the process that can then serve as input to planning and control systems. From these insights, the following questions arise.

Research Questions - Digital Agriculture

RQ 1 How can we derive process information about an ongoing agricultural logistic process?

- RQ 1.1 Which methods of data analysis can extract process information from machine telemetry?
- RQ 1.2 Which additional data besides telemetry is necessary to derive process information?

RQ 2 How can we work with process information in terms of explicit formal knowledge?

- RQ 2.1 How can we semantically model the domain of agricultural logistic processes?
- RQ 2.2 How can we enable automatic inference and rule-based reasoning about the processes?

To approach these agricultural subjects, we draw significantly from the scientific fields of semantic technology and robotics. In this context, the thesis concerns the subject of semantic maps and the development of spatio-semantic reasoning to address the following problem.

Problem Statement - Semantic Mapping

Semantic maps add to classic robot maps spatially grounded object instances anchored in a suitable way for knowledge representation and reasoning. They enable a robot to solve reasoning problems of geometrical, topological, ontological, and logical nature in addition to localization and path planning. Recent literature on semantic mapping lacks effective and efficient approaches for grounding qualitative spatial relations by analyzing the mapped entities' quantitative geometric data. Nevertheless, such qualitative relations are essential to perform spatial and ontological reasoning about objects in the robot's surroundings.

Therefore, methods of applying quantitative spatial analysis in combination with formal semantic reasoning to further the applicability of semantic maps in application domains that specifically benefit from inference over derived spatio-semantic knowledge should be explored.

Research Questions - Semantic Mapping

RQ 3 How can semantic information about entities in their environment be derived by analyzing their spatial relationships?

- RQ 3.1 How can we ground spatio-semantic information?
- RQ 3.2 How can we implement a grounding mechanism in the context of semantic maps?
- RQ 3.3 How can spatio-semantic reasoning be applied in a real-world domain?

1.2 Scientific Contributions

This thesis addresses the above research questions and contributes to semantic mapping and the topic of monitoring agricultural processes in four parts.

First, spatio-semantic reasoning and the benefits of including spatial database and dedicated geometric analysis to semantic maps are discussed and demonstrated. This is done without an agricultural application in mind but in the context of mobile robotics.

Second, a semantic model for agricultural resources and processes is presented, which is capable of representing agricultural machines, their environment, and the activities and tasks that have to be performed towards a particular goal. It is shown that opting for a formal, ontological description of concepts and their relations is beneficial, because it allows for automated logical inference over acquired facts and facilitates growing a knowledge base about the agricultural application domain.

Third, the principles of spatio-semantic reasoning are applied to the agricultural domain. It is demonstrated how spatial analysis can be effectively coupled with semantic reasoning and how grounding and analyzing spatial and topological relations over time can uncover knowledge about the work process previously covert in the machines' telemetry data. A prototypical software system that is capable of monitoring the process states of agricultural machines during process execution is presented. The implemented monitoring system provides a qualitative classification of the process states and quantitative evaluation of associated key performance indicators, both in accordance with the presented semantic model.

Fourth, an example for a digital assistance application is provided that was realized based on the developed process monitoring capabilities. It illustrates how online process analysis can aid in tracking digital service contracts and service level agreements and automated documentation and accounting for harvesting campaigns.

1.3 Structure of the Dissertation

This cumulative dissertation consists of two parts.

Part I provides a summary of the presented research. Chapter 1 introduces the topic of this thesis, states research questions to be addressed, and summarizes the thesis' scientific contribution. Chapter 2 discusses the key contributions of this thesis in the context of the given research questions. It provides a review of the state of the art in related digital assistance systems, followed by a summary and discussion of the content of the included articles. Chapter 3 summarizes the results of this dissertation and gives an outlook on related future research topics.

Part II comprises the included articles of this dissertation. All articles were orderly published in scientific journals or conference proceedings. The bibliographical information is provided at the beginning of Part II; so are the statements regarding the (co-)authorships. The included article's contents were reproduced as published. As a result of this, the articles in Chapters 4 and 7 are in German, while those in Chapters 5 and 6 are in English. For enhanced readability, all articles were reformatted to align in appearance. Font styles and text sizes were unified, and figures and tables were continuously numbered throughout all publications. Furthermore, a uniform citation style was applied, and all references are consolidated at the end of the thesis.

Chapter 2

Digital Assistance for Agricultural Logistics

This chapter discusses the contributions of this thesis in the context of the above research questions. It highlights the key concepts and results of the included publications and argues how they improve digital assistance systems for ongoing agricultural processes.

The remainder of this chapter is structured as follows.

Section 2.1 presents the agricultural context of this thesis. It discusses the challenges of agricultural logistics and highlights why operational decision-making and the ad-hoc decisions that come with managing ongoing processes and coping with unforeseen situations are vital in the context of agricultural decision-making. It also defines what agricultural logistic processes are and introduces the process of forage maize harvesting in detail, as it will be the example process used throughout the thesis. The section closes with reviewing the tasks and business processes related to managing an agricultural harvesting campaign.

Section 2.2 concerns digital assistance systems for agricultural logistic processes. It reviews the current state of the art in planning, monitoring, controlling, and documenting agricultural processes, focusing on solutions applicable to supporting agricultural logistic work and related decision-making and business processes. Based on our review, we discuss open topics and derive essential requirements for future digital assistance solutions.

Section 2.3 directly addresses some of the identified open topics and relates them to this thesis's scientific goals and research questions. The section introduces the core concepts of online process analytics to answer the posed research questions and explains why online process monitoring is essential to implement support features that aid agricultural practitioners during ongoing agricultural logistics processes. It motivates our spatio-semantic approach to realizing online process monitoring and introduces the target analytics functions and features to exemplify our concepts. It, thus, provides the broader conceptual context that ties together the individual contents of the included articles.

Sections 2.4 to 2.6 then summarize the contributions of the included articles to this thesis' overall topic and provide details on the prototypical implementation of our process monitoring system for forage maize harvesting and the automated service tracking and accounting assistance.

2.1 Challenges of Agricultural Logistics

2.1.1 Agricultural Decision Making

To manage a successful and sustainable agricultural production business requires making decisions on how to utilize the available resources best and align long-term strategic decisions with short-term decision-making correctly. Since agriculture is highly interdisciplinary, making an informed decision requires integrating information about many knowledge domains and applying them to a particular business's situation. In general, effective agricultural management always entails coping with external disturbances since agriculture is situated in highly dynamic environments, i.e., volatile economic market or changing weather conditions due to climate change and addressing unforeseen implications, e.g., machines breaking down.

Hence, decision-making in agriculture is a challenging task for agricultural practitioners. Making the right decisions of utmost importance for the agricultural business, and several studies have shown that solid managerial skills are a better predictor of a farm's long-term success than the available resources [153, 192]. Agricultural decision making is widely studied [75, 97, 143, 177] to better understand how agricultural practitioners decide and whereby agricultural decision making can be improved.

This research is vital in the context of designing digital assistance systems for the agricultural sector, too, since it determines the scope of agricultural decision making, i.e., who decides over what and under which constraints and challenges. It is essential to consider these factors to determine how digital systems can be integrated into agricultural practitioners' decision process, inform and support their decision making, or even (partially) delegate it to autonomous systems.

2.1.1.1 Managing Agricultural Production Systems

Regarding the context in which agricultural decision-making occurs, Martin-Clouaire and Rellier provided a helpful model. In [143], they abstractly described an agricultural production system (e.g., a farming business) as consisting of three sub-systems: the management system, the operating system and the bio-physical system, which interact with each other. See Figure 2.1.

The *bio-physical system* refers to the natural resources (e.g., arable land, crops, livestock) and the biological processes that produce the output of the agricultural process. The *operating system* refers to the collection of resources and processes used to execute the agricultural work that goes into the biophysical system's cultivation. It includes natural, artificial, and human resources, i.e., a farm's agricultural machinery and workforce, the agricultural processes, and the materials that treat the bio-physical system, e.g., natural and chemical fertilizers, herbicides, and pesticides. Finally, the *management system* refers to the managerial instance of the agricultural production system, which decides how to apply the operating system to cultivate the bio-physical system. This term usually refers to the farmer and other agricultural practitioners tasked with management responsibilities but may include digital assistance or automation within the decision-making process, too.

In the context of this model, agricultural decision-making is concerned with the internal operations of the management systems that gather information about the other system components as input, which is then thought over and turned into decisions about how to use the operating system. The management systems' responsibility is to utilize the operating system as effectively

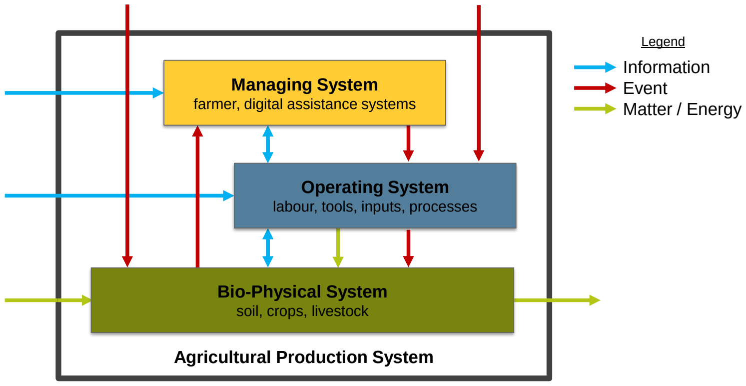


Figure 2.1: An agricultural production system (e.g., a farming business) can be viewed as three interconnected sub-systems: the bio-physical system, the operating system and the decision system and all uncontrollable external influences as fourth component. Figure reproduced and adapted from [143].

as possible to cultivate the bio-physical system, which ultimately produces the marketable output of the agricultural production system.

One can distinguish between the different stages through which each (set of) agricultural operations go through. On a coarse level, one can differentiate between a *planning phase* in which agricultural operations are planned, scheduled, and prepared and an *execution phase* in which agricultural work is actively done [43, 195]. These phases are followed by a third *evaluation phase*, which concerns the assessment of the progress and quality of the work during and after its execution. See Figure 2.2 for an illustration of the (sub-)phases and their interdependencies.

The *planning phase* can be distinguished into a production design phase and a scheduling phase. During the design phase, a top-down approach is taken, in which production goals are determined in terms of targeted and expected results. Likewise, a set of system components is selected for the production of these goals. This phase constitutes a high-level alignment of the business plan with the agricultural production system. During scheduling, the production targets are broken down into specific agricultural operations, which are then brought into a preliminary temporal sequence. With this, the production plan must be aligned with the production system's resource constraints, including temporal constraints, labor supply, production priorities, crop requirements, and available machinery. The scheduling required operations can be broken down into specific work tasks and assignments of resources to tasks.

The *execution phase* can be detailed into three parts. The operating phase constitutes the actual activities of agricultural work performed by and with the resources. The evaluation phase is concerned with monitoring the progress and intermediate results of the ongoing work. The controlling phase is concerned with re-planning and adapting the ongoing work tasks during execution to address deviations from the production targets and other unforeseen events.

Depending on the scope and impact of the decisions made during planning phase, these are often classified into strategic, tactical and operational decisions [104, 176, 195]. *Strategic decision making* is concerned with identifying long-term objectives for an agricultural production business

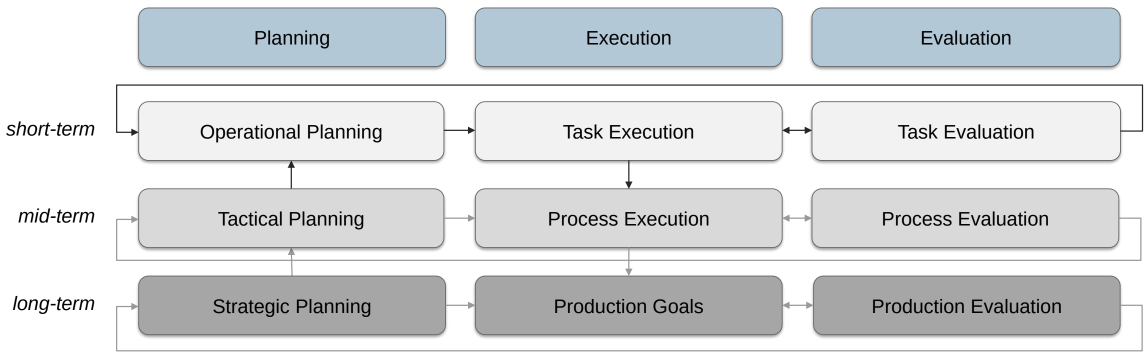


Figure 2.2: Management decisions in agricultural contexts can be structured based on their temporal scope. While strategic and tactical decisions determine the overarching goals and directions of agricultural production businesses, operational decisions are concerned with the actual agricultural work.

and setting goals for the farm that should facilitate a sustainable business over several years. *Tactical decision making* concerns the implementation of strategic decisions in the course of an agricultural season, usually one year, and, thus, related to mid-term planning. *Operational decision making* is concerned with the short-term planning of the specific activities and tasks to bring the tactical plans into execution.

Operational decision-making occurs both during the last steps of the planning phase and during execution. During planning, it is based on the mid-term schedules created during tactical planning and requires breaking down tactical goals into short-termed actions/tasks that constitute a particular agricultural operation or a farm's everyday activities. During execution, operational decision making is concerned with implementing a feedback loop that takes ongoing evaluation results and makes ad hoc decisions about minor or major course corrections in an ongoing activity or changed the way processes are executed between process iterations.

Finally, the *evaluation phase* is concerned with analyzing the progress and quality of the agricultural work and products produced. It takes place both during and after process execution and serves two different purposes. While the process is ongoing, evaluation is aimed at quality control and serves as a trigger for corrective actions. After process execution, it is concerned with process documentation, accounting, and compliance, especially when agricultural work is conducted in the context of a business process, cf. Section 2.1.3.2.

2.1.1.2 Challenges of Decision Making

Agricultural decision-making is complex and difficult since agriculture is inherently interdisciplinary and affected by various sciences and knowledge domains. Fountas et al.., therefore, characterized decision-making in agriculture as "information-intensive" and discussed how agricultural decision-making requires gathering and processing data from these various contexts to inform the decision processes. In [97], they provided a *decision schema* that structures the different natural and digital factors that affect decision making, cf. Figure 2.3 c).

To make the right decisions, an agricultural business manager must relate the *general context* of agriculture and associated fields with the specific *situational context* of his agricultural business. To understand the requirements and limits of the bio-physical system to be managed, knowledge in biology, environmental science, and the respective legal requirements is required. Likewise, managing the operating system demands a solid understanding of economics and market trends and technical expertise in agricultural work processes and agricultural machinery. These domains must be understood and accounted for during decision-making while relating the available knowledge to the agricultural business's particular situation. This includes the overall management strategy, the farm's resources, constraints, and the events and circumstances that trigger the decision-making process. While many decisions are timed throughout the biophysical system's natural cycles, external events may force decisions and influence their outcome.

Anticipating and coping with external influences is one of the main challenges in agricultural decision-making since all sub-systems are subject to external factors, which can not be controlled. Examples for such disturbances in the mid-term to long-term may include: The bio-physical system being affected by changing climate conditions varying the growth of crops or pests that inflict and affect crops. In this case, the management system may need to adapt its strategic approach or invest in newly available technology or knowledge that may open up new means of adaption. In the short term, disturbances of the bio-physical system may include changing weather conditions, e.g., heavy rainfall changing the trafficability of the field. In contrast, the operating system may be affected by an agricultural machine breaking drown or an employee is sick. In these cases, the management system needs to formulate immediate responses or preventive measures.

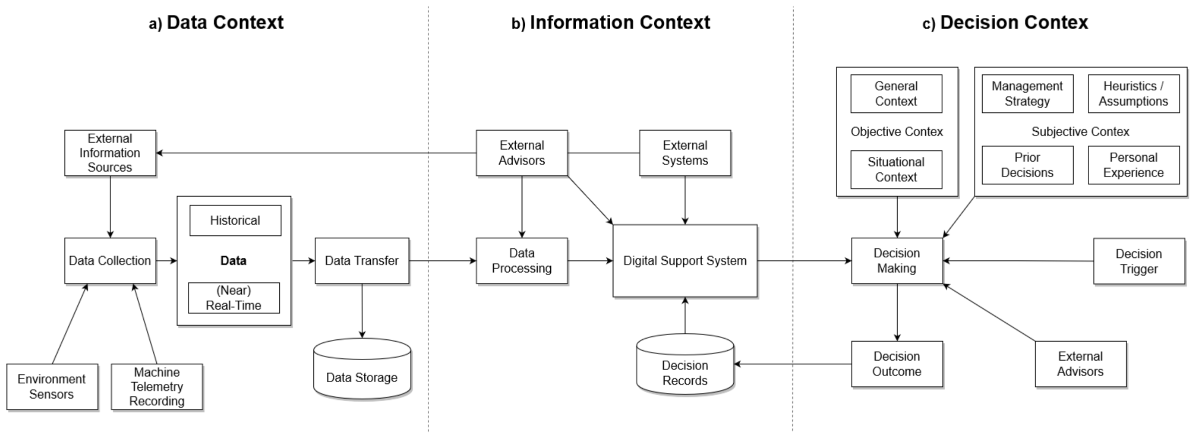


Figure 2.3: The *decision schema* by Fountas et al. shows the different contexts that go into agricultural decision making and illustrates how data collection and processing can enable digital support systems to inform human decision makers. Figure reproduced and adapted from [97].

In a data-driven approach to agricultural decision making, the gathered contextual and situational information can deduce potential options during decision-making and help decide for a good outcome. Ideally, a farm manager would make all decisions based on the relevant facts and weigh different alternatives based on their potential outcomes, short-term and long-term consequences, and choosing for the optimum. However, due to the complex set of domains involved, having full knowledge of a decision's context is intractable. A human decider can not gather or process all the relevant information affecting both the bio-physical and operating system. Therefore decision-making is often based on heuristics rather than derived from quantitative analysis and comparison of potential actions. On the contrary, agricultural decision-making is often informal and qualitative and utilizes past experiences and analogous reasoning. Every decision is influenced by the farmers' personality, critical assumptions, and subjective perception of past decision outcomes.

Therefore, Day et al. characterize agricultural decision making as *tacit*, meaning it is based on implicit knowledge, experience, and intuition [75]. They state: "The farmer's decision process is more consistent with the bounded rationality perspective than with the omniscient optimizing one". Similar assessments can be found in [97, 104]. Of course, the severity of this assessment varies from decider to decider. It may be overly critical regarding cases of strategic decision making, where time for research and consideration is plenty. However, regarding operational decision-making, the assessment is admissible since in this case the time to research and evaluate is limited, and a decision must be made ad hoc, even improvised.

These are challenging constraints under which agricultural decision-making takes place. The situation gets aggravated further by the fact that multiple actors usually conduct agricultural processes both in terms of *process participants* and *business partners*. As a result, decision-making usually can not be centralized onto one designated manager but is dispersed across all involved individuals, even though this is desirable to ensure that all actions lead towards a common goal. This holds especially true for agricultural processes with spatially dispersed sub-processes, such as harvesting, since keeping central oversight is especially hard to realize. Paired with high variance in process understanding, due to a mixture of skilled agricultural professionals with untrained agricultural assistance workers, ensuring *decision consistency* is a crucial challenge.

For operational decision-making, in particular, it is apparent that during process execution, the executing workforce makes small and local decisions for the seeming benefit of a sub-task, which are not necessarily in line with optimizing the overarching process goals. However, at the same time, quantifying these effects and their potential damage is hard for the process managers since the data for evaluation is usually not available, at least not during execution.

Through the digital transformation of agriculture, data has become an essential input into the decision-making process, too, and digital systems have been proposed as a helpful tool to support agricultural practitioners in their efforts to collecting contextual information, extract relevant insights and make the right decisions, cf. Section 2.2.

The *decision schema* depicted in Figure 2.3 a) to b) illustrates this data-driven approach to the agricultural decision process. To inform the decision process, a farmer can use a digital system, which can provide relevant data about the decision's context, enumerate potential options, and information about the expected outcomes and consequences. The digital system relies on data collection, management, and processing techniques, to provide these insights. The data collection steps include collecting relevant information sources, including agricultural telemetry data, environment sensors, and weather forecasts, and other external sources, such as scientific literature or expert knowledge provided by third-party advisors. The collected data is then transferred and stored in database systems to be available for the processing step. The data that goes into processing can comprise historical records (including past decisions and their outcomes) and very recent data on the current situational context, which is collected and processed in near-real-time. The available information must be filtered and applied to the decision problem at hand through data processing techniques. The results of this process, valuable insight can be generated and provided by the digital support to the decider.

The decision schema is meant to provide a general framework on which digital assistance and decision support can be realized for many agricultural processes. It aptly captures the common aspirations towards digital assistance systems in agriculture, some of which are already sufficiently addressed in state of the art, some not, cf. Section 2.2.

In the context of Martin-Clouaire and Rellier's conceptual model of agricultural production systems, a digital system exhibiting such features can be understood as a sub-system component of the management system. It is a tool that informs and assists a human decider in his decision-making and aids in carrying them out via the operating system. While the term originally referred to the legal entity responsible for an agricultural production business (e.g., the farmer), a broader definition of the management system allows multiple entities to be involved in the decision-making process. This includes delegating operational business and deciding to a foreman or using a *digital assistance systems* to augmented human decision making. Likewise, a (partial or total) delegation of decisions to an *digital automation system* that manages parts of the agricultural production system without human oversight would fit this definition, too.

2.1.2 Agricultural Logistic Processes

Logistics is an essential aspect of agriculture. Almost all processes in crop production include some material flow, either from the farm to the field for cultivation or from the field to the farm or other locations relevant for post-processing. We define *agricultural logistic processes* to be all those processes that include transport of agricultural goods in the context of a complex agricultural production process, to stress that transportation is a sub-process supporting the primary production task. In this notion, agricultural logistics differ from classical logistics.

The *transport of agricultural goods* can be defined as a sequence of loading, shipment, transshipment, and unloading operations to bring a load of agricultural goods from a specified origin, e.g., the field, to a target destination, e.g., the farm, within a certain time frame [51]. These operations are executed either by a dedicated transport vehicle or other agricultural machines that allow for material storage, e.g., a seeding machine.

The transported goods can be in bulk (e.g., maize chaff) or pieces (e.g., potatoes in a box). They can be loaded from to a storage unit or be loaded from an agricultural machine onto another. The latter is expected since many agricultural goods are produced during agricultural work by agricultural machines, such as harvesters, and are either directly loaded onto a transport vehicle or temporarily left in the field to be picked up later on, e.g., during baling operations. Similarly, transloading operations from one transport vehicle to another are commonly seen. Transshipment operations, i.e., usage of an intermediary storage location, are also common. For unloading, loading off to static storage or a dynamic post-processing infrastructure is common.

Tractors with attached trailers are the default for transport vehicles in the agricultural domain. Depending on the agricultural tasks and logistical requirements, different kinds of specialized transport trailers exist. General-purpose trucks and dedicated transport vehicles, such as agricultural trucks, are used, especially for long haul transports [91].

Examples for agricultural logistic processes are the harvest of crops, such as grain or maize, the retrieval of bales after green-land harvest or seeding operations, and other cultivation measures.

In [193], Sørensen et al. present a systematic approach to differentiate between different kinds of logistic activities in the agricultural domain. A fundamental division can be made based on the direction of material flow. All operations that produce agricultural goods as their output (e.g., harvesting crops) can be classified as outbound logistics. In contrast, all processes that support agricultural goods production through material input (e.g., seeding and fertilization) are classified as inbound logistics. Further differentiation lies in the origins and destinations, as well as the applied means of transportation involved.

In-field logistics describes the operation within one particular field ranging from single machine applications to multi-machine applications and include infield transportation. *Inter-field logistics* describes operations within a set of fields and include transportation between fields via roads and agricultural paths. *Inter-sector logistics* describes operations within a local scope or sector, including fields around a farm, potentially other important waypoints. This mode of logistics regularly includes transportation by road. *Inter-regional logistics* describes operations across more considerable distances, e.g., when agricultural goods are handed over to other industry branches In this case, transportation by all means of transportation, including highways, rail- and waterways, is within scope.

Because of the vast areas cultivated by agriculture and the large amounts of agricultural goods produced, logistics constitutes a lot of the agricultural work. For Germany, as an example, it is estimated that transport activities make up between 40 % to 50 % of the agricultural outdoor work [29]. According to the DBV Situationsbericht 2014/2015, about 500 Mio. t of biomass were transported in 2011 alone [81]. Compared to other logistic industries, Germany's agricultural sector's logistical effort is equal to that of the rail transport segment, but the average transport distances are minuscule compared to those in classical logistical sectors [29]. This is because most transport activities are local or regional, not inter-regional, leading to a disproportion between transport volumes and distances.

Transport challenges in agriculture are also steadily increasing. These are the reasons:

Crop producers are very inclined to choose particular crop types, to maximize yield under their regional conditions and economic benefits. This results in a concentration of the same crop type being cultivated in a particular area. Due to similar growing conditions, these crops exhibit very similar growth such that fertilization and harvesting operations across all these areas must be executed in very narrow time windows. The workloads for farmers and agricultural contractors thus peak seasonally. The increased yields add to the transport volume, too. Both factors significantly increase the demand for well-executed transport operations.

Similarly, the economic stress in the agricultural sectors has led to a decline of small farms and benefited larger farms and farms' mergers. These changes to corporate structures also led to higher degrees of specialization across different locations and, thus, more considerable transport distances. The growing bio-energy sector also leads to an increased demand for biomass production and associated harvest and transport work [30].

In summary, logistics is an essential sub-process in agriculture, and though most agricultural businesses are not focused on logistics per se, the business implications are understood, and controlling logistics costs is the focus and interest of many businesses.

Among the most transport-intensive agricultural processes are harvesting and manuring operations, where large amounts of goods need to be transported from or to the field.

When creating digital assistance and decision-support solutions for agricultural logistics, it is valuable to focus on these operations, as they provide the most potential impact. Under the assumption that inbound and outbound logistic processes share similarities and solutions for one can be transferred to the other, we opted to focus the remaining discussion of this thesis on agricultural logistics within harvesting processes. Hence, we will consider only outbound material flow, where harvested crops from a set of fields are transported to one storage facility. This scopes our investigations to in-field, inter-field, and inter-sector logistics. Aspects of inter-regional logistics and beyond are out of scope. This excludes all interactions with the broader agri-food supply chain, which concerns the delivery of agricultural products to the consumer [216].

2.1.3 Forage Maize Harvesting

To focus the development of a digital assistance system for agricultural logistics, it is helpful to target an exemplary process for which particular support functions can be developed. Within this thesis, we chose the process of forage maize harvesting to guide the conceptual design of online process analytics systems (cf., Section 2.3) and our implementation of selected online process monitoring capabilities (cf., Sections 2.4 to 2.6).

Forage maize harvesting is concerned with the harvest of maize plants for the production of whole plant maize silage, which is primarily used as fodder for ruminants and substrate in biogas plants [83, 98]. Since whole maize plants are harvested, much biomass is produced quickly. Due to the sheer volume of harvested crops, bunkering crops within the harvester is not feasible, as it is commonly done in grain harvesting. Consequently, the harvested material is directly loaded onto transport vehicles and hauled to the silo without a transloading step to decouple in-field transportation from the field-to-silo transport. This constant demand for transport capacity at the harvester makes the coordination of the transport chain especially important and challenging.

Forage maize harvesting is, therefore, a well-suited exemplary process to study agricultural logistics since the overall harvesting process is mainly constrained by the performance of the transport logistic sub-process. Likewise, the demand for support solutions in managing and executing forage maize harvesting processes is high, as discussed below.

2.1.3.1 Harvesting & Logistic Processes

The agricultural work process of forage maize harvesting consists of several sub-processes:

The agricultural logistic process consists of the *harvesting of crops* in a field, the *transport of crops* from the field to a silo facility, an optional *weighing of harvested goods* and finally the *unloading of crops*. The agricultural process is finalized by the *compaction of crops* within a silo facility to enable ensiling, which is (partially) coupled to the logistic process, cf. Figure 2.4.

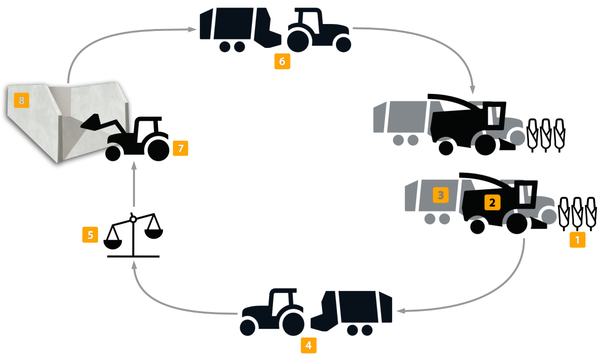


Figure 2.4: The logistics cycle of forage maize harvesting.

Harvesting & Loading The harvesting of forage maize takes place in a maize field ① and is done using Self-propelled Forage Harvesters (SFHs) ②. These harvesters cut the entire maize plants using a cutting unit that cuts at the plant's lower shaft. From there on, the entire plant is mechanically fed into the machine, where it is internally chopped into pieces.

Due to the large quantities of material processed, SFHs do not have an internal bunker and capacity to store material temporarily. Therefore the chopped material is immediately loaded onto a Transport Vehicle (TV) ③ using the harvester's loading boom, to throw the material into the TV's trailer (cf. Figure 2.5 (a)). Hence it is required that the SFH is serviced with a TV for loading at all times. Otherwise, the production process comes to a halt.

The TVs that services the SFH are usually a tractor-trailer combination. The type of trailer varies from simple dump trucks to dedicated forage transport trailers (e.g., scraper floors). To properly service the SFH, usually, a fleet of TVs is deployed. This fleet provides the required transport capacity to bring the harvested goods to the storage facility, i.e., the silo ④. This process of harvesting and loading is conducted until a field is completely cleared. It repeats if additional fields are to be harvested.

Transport, Unloading and Empty Runs Once a TV is fully loaded, the harvested good is transported back to the silo facility ④. The *transport drive* can be separated into its in-field portion and the transport by road, but it is common in forage maize harvesting that both segments are done by the TVs without a transloading at the field boundary. The harvested goods are unloaded at the silage facility.

In the following, we assume that the chopped maize is brought into a bunker silo, where it is built into a pile and prepared to become silage, as this is the most common form of forage maize storage. For *unloading* in a bunker silo, a TV can either discharge onto the silo's forecourt or immediately onto the forage pile, given it is trafficable by the TV (cf. Figure 2.5 (b)). After the material is discharged, the TV returns to the SFH through an *empty run*. This completes a single transport cycle. The transport process is repeated until the harvesting process stops and the TV loaded last reaches the destination for unloading.

Weighing of Transported Goods Before a TV discharges at the storage facility, ideally, its load is determined using a weighbridge ⑦. Weighing is used to determine the exact load of the harvested goods, which is necessary information for documentation and accounting purposes. The net weight of the harvested goods is of final interest. To derive it, the tara weight of each TV is determined regularly, in a process step called *tara weighing*. Given this measurement, it can be subtracted from the TV's gross weight determined during *load weighing*.

If weighing is incorporated into the transport process, it is common to weigh the TVs every time before they discharge. However, weighing is an optional process step. Whether applied or not, it depends on the business processes in which the agricultural work process is embedded. Using a weighbridge provides the most accurate way of documenting the amount of processed materials but not the only option. Alternatively, one can utilize the internal yield measurements of the SFH or estimate the resulting yield by counting transport vehicles or estimating the resulting volume of the silage pile. Using the SFH internal yield monitoring capabilities allows estimating the amount of goods loaded onto a TV during the loading procedure. These estimations can be used for documentation purposes, too, but are subject to measurement errors in the yield meter. To prevent these errors, a regular calibration of the yield meter is advisable. It can be conducted by comparing the estimated load against a proper load measurement obtained by the weighbridge. Again, this calibration process requires a visit at the weighbridge, but not necessarily on every transport run, which makes the process leaner.

Regarding the logistics, visiting the weighbridge introduces an additional way-point to the transport route of the TVs from the field to the silo. Depending on the weighbridge's location, this can result in a detour from the shortest path between field and silo, as it is not given that the silo facility has a weighbridge on site.

(a) Harvesting Crops & Loading a Transport Vehicle

(b) Unloading & Compacting

Figure 2.5: The process of forage maize harvesting is concerned with the harvest of maize plants for the production of whole plant maize silage. (a) Self-propelled forage harvesters are used to cut and chop entire plants. The resulting maize chaff is immediately loaded onto transport vehicles and hauled to a silo facility. (b) At the silo, the material is unloaded and compacted for conservation and storage. Images provided by CLAAS [64].

Compaction and Conservation To properly store forage maize over time, the chopped material must first be built into a silage pile ⑧, which is then compacted and sealed. So-called Compactor Vehicles (CVs) are used to preparing a silage pile.

These move material from the forecourt of the silo facility onto the silage pile, where it is then repeatedly run over to compact the material. The forecourt serves as a material buffer from which the CVs can transfer the material needed to build the silage pile further. Tractors equipped with a shield front attachment and additional ballast are commonly used as CVs . Wheel loaders are used, too. Usually, multiple compactor vehicles work in parallel when the silage facility is large enough or multiple silage pits are filled and prepared simultaneously.

The harvested material is conserved by creating maize forage through an anaerobic fermentation process, i.e., due to the metabolism of lactic acid-forming bacteria. For this to work, the silage pile is meticulously sealed after it is formed and properly compacted [98]. Since the silage quality is heavily dependent on proper compaction, great care must be spent on this work step. If the pile is not compacted correctly, aerobic processes can lead to rot and mold within the silage after it is opened. This may render the silage unusable as feed and decreases its energetic value.

Compacting a silage pile too much is virtually impossible but should be prevented nonetheless because at a certain level of compaction, all further work is excess and leads to unnecessary time and resource consumption. It is therefore vital to properly dose and distribute material across the silage pile before compacting it.

The process of silage pile building and compacting the harvested material is conducted in parallel to the harvesting and transport process since TVs can discharge at the silo's forecourt regardless of the ongoing compaction work. The compaction process is, hence, partially decoupled from the agricultural logistics process of harvesting and transporting. However, the logistics must be coordinated to be in close harmony with the compaction and storage process's progress. Since creating silage of high quality is the primary process objective of the entire process chain, and proper compaction is crucial for successful conservation, the downstream process is ideally considered the driver of the process chain and determines the target production and transport performance of the upstream logistics.

Balancing Production, Transport and Storage From a logistics point of view, the harvesting process can be understood as the production and transport of bulk goods with the compaction process as the subsequent storage of the material. In this regard, agricultural logistics resembles classical logistics, but there are notable differences.

The most crucial difference is that the environment in which the process takes place is a cultivated but natural environment that produces inherent variants in the process inputs due to variants in the crops. Considering the harvesting process as the *source* of the logistical network, the yield on the field together with the work performance of the harvesters determines the overall production rate and material flow into the network, measured in *produced* tonnes per hour.

Likewise, the transport process can be understood as the flow of material through the logistical network. Here the transport capacities and speeds of the active transporters determine the overall transport rate, which can be measured in *transported* tonnes per hour. The location of the field and silo and the resulting transport distance play an essential role in the design of transport logistics. The available route network and the permitted speeds and factors such as trafficability of natural terrain also affect the transport performance. Usually, driven routes

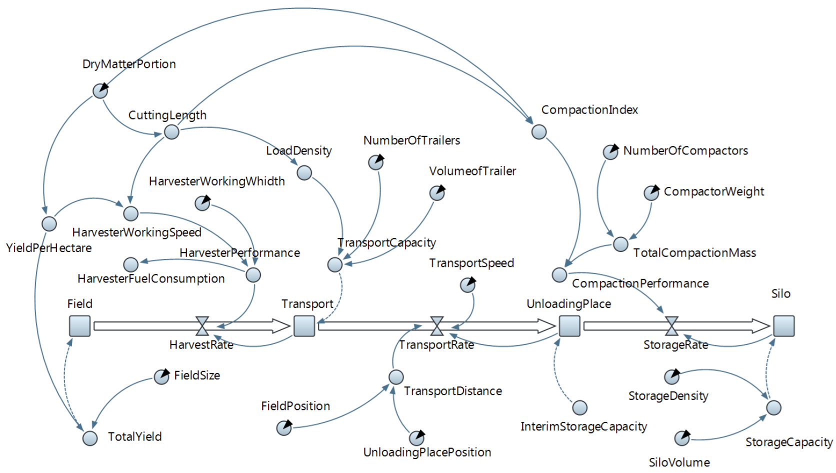


Figure 2.6: The different controllable and uncontrollable influence factors on the forage maize harvesting process. Figure reproduced from [201].

are not as advantageous as in other logistic scenarios, e.g., warehouse logistics, and commonly shared with other non-process participants, similar to classical road-bound logistics, but with the disadvantage of more rural roadways, such as being narrow.

Finally, the silage building process can be considered the sink of the logistics process. Again the performance of the compactor vehicles determines the rate of *stored* tonnes per hour. As the harvest performance, the compaction performance is dependent on the natural properties of the harvested material. For example, the dry matter portion of the maize determines how well the material can be compacted and how much work the CV have to exert.

In summary, there are many interdependencies between controllable and uncontrollable influence factors that need to be considered. Steckel provided a comprehensive illustration of these factors and their relationships, cf. Figure 2.6. For an extended discussion of the different parameters, please refer to [201].

To ensure an effective and efficient forage harvest, all sub-processes must be controlled so that the harvest, transport, and storage rates are in alignment. Ideally, the tonnes per hour that flow into the logistical network are balanced and stable through and out of the logistical network. Setting up harvesting operations in such balanced way is a management task.

2.1.3.2 Organizational Processes & Management Decisions

Initially, the forage maize harvesting process appears to be a simple cycle with a manageable number of participating machines and way-points, but it is a complex undertaking in reality. All interactions among the agricultural machines need to be closely coordinated to ensure a balanced work process despite the perturbations of a dynamic environment. However, forage maize harvesting is not only an agricultural and logistics work process; it is also embedded in organizational processes and subject to various managerial decisions and economic considerations.

This section discusses the managerial tasks and challenges of conducting a forage harvesting process under the underlying business-related aspects and constraints. First, it will introduce the concept of a *harvesting campaign* as an instrument to structure and control agricultural work. Secondly, the organizational processes and critical decisions for conducting a harvesting campaign will be introduced following the three phases of *pre-planning*, *execution* and *post-processing*, introduced in Section 2.1.1. Finally, the related business process is introduced with its key aspects of *contract formation*, *documentation of accomplished work* and *billing.* From here on, it is assumed that the harvest is conducted as a service by an agricultural contractor for a farmer.

Managing Harvesting Campaigns During a harvest operation, usually, more than one field needs to be harvested. In this case, the work process introduced in the section above simply repeats. The harvester changes location through road travel and continues its work in the following field. Its associated fleet of TVs follows accordingly. The entire harvest operation may continue over multiple days, where the work is suspended overnight and resumed the next day.

A *harvesting campaign* can be understood as the conceptual entity that binds a collection of fields to be harvested, the machines assigned to do the work, as well as the silo facilities to which the harvested goods need to be transported. The affiliation of fields can be based on being owned by the same business entity (i.e., a farm business) or sharing the target location for the harvested crops (i.e., multiple farmers supplying the same biogas plant). It entails the formation of workgroups and the assignments of human operators to machines, too.

We call a set of SFHs and TVs working together on the harvest of one or more fields a *harvesting fleet.* Depending on the field size, it may be that multiple harvesters are assigned to the same fleet, such that they share the work on the field and the transporters, which may serve any of the harvesters within their group. Using multiple TVs is obligatory for such fleets, as discussed above. Even though single transporter “fleets” may occasionally occur, e.g., when the field-to-silo distance is short. Depending on the amount of work, it is also expected that multiple harvesting fleets work in parallel on different fields. In this case, the assignment of TVs to (a group of) SFH(s) is usually kept fixed until there is reason to restructure the division of labor.

Likewise, a *compaction fleet* denotes a set of CVs working together on a silage pile’s formation and compaction. Depending on the circumstance, multiple silage facilities can be worked on simultaneously by one compaction fleet. Having multiple compaction fleets may occur, too, if the silo facilities are not located close to each other.

Managing a harvesting operation is about forming and controlling the fleets of a harvesting campaign, and a set of critical tasks of related decision-making can be identified.

In [43], Bochtis et al. identified several critical tasks for managing fleets of agricultural machinery: capacity planning, task times planning, fleet formation and scheduling, harvesting strategy and route planning, and performance evaluation. These tasks span the various management phases and levels discussed in Section 2.1.1. Since we are most concerned with the harvest’s actual execution, we limit our discussion to tactical and operational tasks.

Capacity & Task Times Planning In the context of harvesting in general, capacity planning is both a strategic and tactical decision-making task. The decision of how many crops need to be cultivated and harvested for an agricultural production system is part of the business’s strategic decision-making. It sets the global targets and boundaries for the harvesting campaign.

The decisions are mainly tactical and concerned with the harvesting campaign’s overall setup regarding the work process. The manager has to decide how many and which machines have to participate in the harvesting campaign. With this, the alignment of the production, transport, and storage capacities is of foremost importance to create a balanced logistical network. On a theoretic level, the total average production performance of all SFHs should equal the total average transport performance of the transport vehicles. The same holds for the total storage performance of the CVs .

Ideally, the chosen setup allows for some spare performance and buffer capacities to adapt to varying field-farm distances and leave room for control. Usually, a trade-off must be made based on the available machinery and workforce, and unforeseen events, e.g., a machine breaking down, may require operational revisions during process execution.

Task time planning is also a tactical task that occurs during pre-planning. It concerns the global schedule of the harvesting campaign and allocates all required resources for a time period of multiple days in which the harvesting shall occur. It is usually preliminary since external constraints such as crop ripeness and weather need to be considered and may lead to frequent re-planning. Nevertheless, it is vital to organize all involved business parties when the harvest is done as contract work, and compromises must be made.

Fleet Formation & Scheduling Once the harvest nears, the next managerial task is to form fleets and schedule the harvesting campaign's fleets. The available resources in the form of machines and personnel have to be allocated to different jobs. Here the previous considerations regarding capacity planning are broken down into assignments on a fleet level, i.e., determining the set and order of fields for a harvesting fleet. This entails creating a schedule, too. It describes when particular harvesting jobs shall be carried out and in which order the transport vehicles should drive from location to location. The field-farm-distance is a commonly used key metric to group fields into clusters since this stabilizes transport performance and helps to balance the material flow from field to farm.

The assignment of a suitable group of TV to a SFH is usually designed to remain stable within one day to harvest day, to prevent frequent re-planning of fleet setups and job assignments. But it may be subject to change, if necessary. Hence, these tasks are all considered operational, as they occur during pre-planning plus need continuous revision during execution.

Harvesting Strategy & Route Planning During the actual harvest, two additional planning problems need to be solved.

The term harvest strategy planning refers to the concrete steps of approaching a field with the harvester. It concerns segmenting fields into distinct parcels that shall be harvested to plan geometric routes to cover the fields. The decision-making is here lead by the overall environmental conditions and temporary constraints, such as designated field entry and exit points, obstacles at the field's border or within the fields. A common practice is to harvest the headland along the entire field to create space for turning the harvester, making in-field travel easier for the transport vehicles, and then harvesting the remainder of the field following straight tracks to cover the field. In practice, though, one can observe that SFHs often diverge from strict patterns to better account for the transport vehicles' load capacities and their ability to drive to and from the harvester. This is due to a high degree of freedom in the SFHs movement. Given their specific plant row-independent headers, they are not restricted to following the crops in a specific way and can load onto transport vehicles from both sides and the rear.

Similarly, route planning between the different waypoints of the transport cycle must be done. Especially for transport vehicles, the question of "Which route is most suitable from farm to the field (and vice versa)?" must be adequately answered. Besides typical constraints such as required drive times and traveled distance, considerations regarding the chosen route's trafficability are essential. Narrow roads and obstacles need to be accounted for, and interference with regular traffic must be minimized. This leads to numerous constraints, primarily due to the vehicle's size. It is common for machine operators to rely on multi-purpose route planning tools (e.g., Google Maps) and knowledge about their local environments.

Of course, all decisions are tightly coupled and directly influence the already made plans regarding a balanced material flow. Hence they must be addressed under the constraints of the targeted capacities and schedules and provide meaningful feedback to the problems above.

Both problems can be addressed as tactical problems during pre-planning, but they are usually left to the operators during execution in practice. As operational problems, they are commonly solved through the operator's expertise and intuition. This is a frequent source of inefficiencies since, especially for contract workers, the details about the local environments are not known. Similarly, it is expected that transport fleets are made up of machine operators from both farmer and contractor, which may lead to problems in communication or misalignments of what is considered best practice.

Performance Evaluation The final consideration for managing a harvesting campaign is the performance evaluation. It concerns monitoring and analyzing the progress and performance of the overall campaign and the individual process participants.

Its purpose is to revise the current mode of operations and evaluate if updates to past decisions are necessary to maintain and improve the ongoing process and ensure successful completion of the campaign. Hence it is an integral part of all the decision-making. It functions as the trigger that starts re-planning the harvest campaign's tactics and provides ad-hoc feedback for changes in the ongoing operations.

Of course, performance evaluation is also related to strategic decisions. Summarizing the campaign's overall performance post-harvest provides valuable input for planning future crop cultivation and harvesting processes.

Several subjects are of concern during evaluation. The primary objective is to ensure that the harvest produces high-quality agricultural goods while being cost-effective for the business. Secondly, it measures the logistical network's inputs and outputs and evaluates the balance of mass flow in the transport cycle. Finally, it is also about adhering to rules and regulations.

How the performance is evaluated depends on the targets and preferences of the agricultural business. Ideally, the evaluation is based on key performance indicators and allows for quantitative measures and assessment, but in practice, it is common to rely on tacit decision-making.

In summary, managing a harvesting campaign poses a set of several planning and control problems, which are all highly interlinked. Additionally, there is a high emphasis on operational decision-making. This is partly because multiple moving machines need to be closely coordinated in real-time, which is challenging. However, it is further complicated by a set of challenges imposed by the highly dynamic environment, leading to increased complexity in the harvest's "decision space".

Crop Ripeness For optimal development of the maize silage, the dry matter percentage of the harvested maize is essential. It should lie in a range of ≈29 to 34 % [83, 98]. Therefore, matching the optimal point of crop ripeness is desirable but hard to achieve for all fields within a particular harvesting campaign. Since the crop's maturation process is dependent on multiple factors, e.g., crop variety, soil conditions, cultivation method and micro-climate, and weather, the state of crop ripeness may vary significantly even within the same or across adjacent fields. Therefore, the agricultural practitioners are tasked with considering ripeness on a per-field basis and finding trade-offs while solving the task planning and scheduling problems.

Short Time Windows Related to crop ripeness, the overall time window for harvesting is usually short. In Europe, all forage maize harvesting is done in a few weeks, from the end of August to the beginning of October. Therefore, farmers and agricultural contractors are under pressure to harvest as many fields as possible in short amounts of time, as long as the crop is in optimal condition. For agricultural contractors, this is challenging since they often have to simultaneously support multiple customers in their harvesting efforts. This places significant constraints on all tactical decisions, especially task time planning, such that compromises are necessary and sub-optimal initial conditions may occur.

Machine Breakdowns & Human Error Correlated with the short time windows comes an increased stress level and overtime for all involved personal. It increases the likeliness of errors during task execution and related operational decision-making. Similarly, the machines are under constant mechanical stress, too. Without enough time for maintenance, it is also possible that machines break down during a harvesting campaign, resulting in forced pauses of varying duration depending on the damage and required time for repairs.

Weather Dependency Another factor is the weather. While wetness in the field due to rain has no direct effect on the crops, it may affect the silage quality and may lead to muddy ground and decrease the heavy machines' trafficability. Depending on the intensity, it makes sense to stop harvesting when the field is too wet or not to start when heavy rainfall is imminent. Since a change in the weather does not affect all fields alike, it does not necessarily mean a complete stop for the entire harvesting campaign. Often changing the field order assignments of harvesting fleets or forcing an acceleration of the current field's harvest (at the expense of quality and energy) can be used to adapt. The weather situation is therefore actively monitored and included in tactical and operational planning at all times.

Other Environmental Uncertainties Harvesting processes are characterized by high environmental exposure and, unlike other manufacturing industries, can not be completely shielded from external influences. Instead, they happen embedded in both natural and human-made environments. Transport vehicles, for example, share the same road networks as regular traffic, and, similar to normal freight transport, traffic congestion can be a limiting factor. Similarly, obstacles in the field (e.g., hiding animals or playing children) may interrupt the harvest.

2.1.3.3 Associated Business Processes

Depending on the area to be harvested and the amount of material to be compacted, conducting a harvesting campaign with a single farmer's own machines becomes infeasible, even for larger agricultural businesses. Hence conducting a harvesting campaign is often bought as a service provided by an agricultural contractor or performed within an agricultural cooperative through sharing machines and workforce.

The following summary addresses the business-related processes of contract formation, joint decision making, service documentation, accounting, and billing. An extended discussion of business-related aspects can be found in Chapter 7 and the related summary in Section 2.6.1.

Contract Formation Before a harvesting campaign can be conducted as a joint operation between a farmer and an agricultural contractor, both parties need to come to terms.

At the beginning of the contract formation, the farmer has to declare his demand for the contractor's service. This includes providing information about the fields to be harvested, the silo facilities as destinations, preferred date or time frame, et cetera. Based on this data, the contractor can conduct a process-oriented pre-planning and allocates the required machines and personal. He then answers the farmer with a cost estimate that defines the mode of service billing, too. Billing can be based on hourly rates and wages for machines and operators or fixed prices per harvested area or compacted mass of silage. Mixed pricing models are standard, too. If the farmer accepts the contractors' offer, a contract is formed on the basis of this agreement.

Since the optimal time for harvesting is crop growth and weather dependent, usually only a coarse time frame is defined as a target. Final timing is then done immediately before the harvest starts. This allows for last-minute changes to the service requirements regarding machines and staff. These changes are usually not reflected in the contract but are discussed informally.

Joint Decision Making & Competing Business Interest During the harvest, both the farmer and the contractor act in their respective managerial capacity. While it is expected that the majority of harvest campaign management is delegated to the contractor's foreman, both business parties are usually involved in the decision-making. With this, it must be taken into account that both parties act both *cooperating* and *competing*.

The farmer is interested in the highest quality of the resulting silage for the lowest price, favoring a slower and careful approach to harvesting and silage compaction. The contractor, on the contrary, has an incentive to speed up the harvest and thereby potentially decrease the product's quality as far as possible without hurting the relationship with the customer. This is because he has to maximize the number of services he can provide to all of his customers to maximize his profitability and refinance the expensive fleet of agricultural machinery.

To account for both parties' business interests, both must participate in the ongoing evaluation of the campaign's progress and align the tactical and operational decisions.

Service Documentation, Accounting & Billing The continuous evaluation of the unfolding harvest process also directly plays into creating documentation of the harvest process.

Comprehensive documentation provides a complete record for all delivered services to be invoiced under the given contract so that the usual business documents, e.g., delivery bills and invoices, can be prepared afterward. It is, therefore, necessary to record the hours worked by the machines and their operators, as well as precisely list the harvested areas and quantities. It is common to work with lump-sum accounting and fixed prices for particular items. For example, all machines' total fuel consumption is compiled and listed separately in agricultural invoices to apply for subsidies. A detailed recording of the activities and performance of individual machines is usually not mandatory.

During the execution phase, the documentation of provided services and harvested goods is primarily the machine operators' responsibility. Modern SFHs can also record their working hours and fuel consumption by machine and print them on a receipt using a cash register printer. However, these are usually collected manually and then passed on to the contractor's accounting department. The operators are also responsible for recording the total weight of the TV measured on the weighbridge.

Given the long and hectic harvest days, missing or incorrect data is oft seen with harvest documentation, which can rarely be reconstructed or corrected afterward, mainly due to the extended periods between harvest and accounting. Usually, the practical day-to-day business is brought forward first, as long as the weather is good and the invoicing is postponed to the period after the harvest season.

In summary, managing an agricultural harvest is all about effectively managing a fleet of agricultural machines and its agricultural work under the agricultural production business's overarching goals. To successfully manage both the harvest in the field and the transport and compaction of material, the agricultural practitioners have to solve numerous interlinked planning problems and account for changing conditions regularly to control and optimize the harvest process. It is also apparent that operational decision-making and ad hoc adaption take a dominant role in harvesting processes. This holds for all harvesting operations in general but is especially true for forage maize harvesting with its strong dependence on just-in-time logistics.

Reviewing the overall approach to decision-making during the harvest's execution, one can identify several key differences between agricultural logistics and classical logistics. In classical logistics, it is common to employ dedicated personal tasked with the disposition, scheduling, and routing of transport vehicles. Most of the decision-making, planning, and optimization are therefore solved centrally in a top-down fashion. In this regard, agricultural logistics differ significantly from classical logistics since its management is usually not based on one central decider. On the contrary, the tactical and operational decision-making is distributed across all the process participants to varying degrees, which is partially beneficial and decremental at once.

It is positive when it decreases the response time to unforeseen events and requires a high degree of expertise for every involved human decider, which is not guaranteed for all machine operators. Likewise, multiple, locally optimized decisions may lead to actions that globally contradict each other and the campaign's overarching goals. Even though there is usually a foreman or business owner involved in managing the overall process, it is often difficult to overview the entire process chain since all participants are spatially distributed, continuously moving, and changing states and performance.

The lack of central oversight can be identified as one of the significant drawbacks in the current state of conducting harvesting processes. Therefore, the need for digital support in the disposition of transport-logistical processes in the forage maize harvest is high. A digital system capable of aggregating information so that both the executing workforce and involved business partners can gain knowledge about the progress and effectiveness of the ongoing harvesting process should be of great benefit. The development of such a system is a worthwhile addition to the set of digital agricultural assistance systems if it can assist human deciders in their decision-making towards controlling and optimizing the process.

To further clarify the requirements towards such a system, we will continue with a discussion of the state of the art of digital support solutions for agricultural logistics processes.

2.2 Digital Assistance Systems for Agricultural Logistics

Digital systems that are supposed to assist agricultural practitioners during their work have been developed for several years. Different kinds of systems with a broad scope of function and application have evolved. We continue with a summary of the current state of digital support systems in agriculture, focusing on agricultural logistics. We start with definitions to distinguish several kinds of digital assistance systems involved in agricultural decision-making.

A general definition was given by Rose et al.:

> Decision support tools, usually considered to be software-based, may be an important part of the quest for evidence-based decision-making in agriculture to improve productivity and environmental outputs. These tools can lead users through clear steps and suggest optimal decision paths or may act more as information sources to improve the evidence base for decisions. – [178]

The definition highlights the importance of evident-based decision-making and stresses that dialog a between the digital support tool and its users is beneficial. It also points to differentiation into either an information source or an active system that processes data to prepare decision options and guide them through the decision process.

In [193], Soerensen et al. introduce Management Information Systems (MISs) as the "information management methods tied to the automation or support of human decision making". They relate MISs to general Enterprise Resource Planning (ERP). While ERP concerns all management activities which support all essential business and processes within the enterprise, Information Systems (ISs) address the overall planning and control activities covering the application of humans, technologies, and procedures of the organization. In this context, a group of MISs can be understood as sub-systems of an IS, which are tasked with analyzing and controlling specific operational activities in the organization. Hence they are "often tailored to the automation or support of human decision making". Referring to [93], the definition also points to the identification of Key Performance Indicators (KPIs), as an important feature.

The definition provided by Soerensen et al. also fits nicely into the model by Martin-Clouaire and Rellier. The management system is tasked with doing ERP-related work and uses an IS that organizes and processes all information about the operating system. Therefore, a MIS can be understood as a part of the management systems that supports human during decision-making.

In the following, we will understand digital support systems for agricultural decision-making to be synonymous with MIS as part of the agricultural production business's management system, cf. Section 2.1.1. To account for the different degrees of how a human decider interacts with the support system, cf. [178], we will further differentiate between three categories of MIS: An Information Support System (ISS)'s primary purpose is to collect and process data into structured and relevant knowledge, such that a human decider can use it to be *informed* for decision making. A Decision Support System (DSS) aids a human decider in the decision process and augments it by providing enumerations of decision options and evaluations of potential outcomes and suggestions regarding optimal decision paths. An Autonomous Decision System (ADS) manages parts of the agricultural business without human oversight and can autonomously make decisions on specific tasks if a human operator delegated these. Of course, digital assistance systems – and especially the products build upon them – may incorporate functions with various degrees of decision autonomy side-by-side, depending on what suits the practitioners needs.

Figure 2.7: Management information systems can be differentiated into information support, decision support and autonomous decision systems based on the degrees of digital assistance they provide.

Next, we present the current state of digital decision support tools available in the agricultural sector. We will begin with a summary of digital systems onboard of agricultural machines and continue with several prominent categories of agricultural digital support systems, including:

Precision Farming Applications (PFAs), which utilize the availability of geo-referenced machine telemetry to improve agricultural in-field processes; Farm Management Information Systems (FMISs), a holistic form of assistance systems that supports the farmer in managing her farm and various kinds of agricultural work; and Fleet Management Systems (FMSs), the dedicated tools for planning and executing processes with fleets of collaborating machines.

2.2.1 Digital Agricultural Machines

Agricultural machines are the primary tools for executing agricultural work. They can be distinguished into self-propelled machines and agricultural implements. While the former covers general-purpose tractors and specialized machinery (i.e., harvesters, system tractors), the latter includes all kinds of attachments to self-propelled machines (especially tractors), including transport trailers and plows seeders, and many more.

Modern agricultural machines are already heavily digitized. The use of the system's CAN-BUS and ECUs for numerous internal control purposes of the machines is standard for self-propelled machines and common among control intensive implements (like seeding machines or fertilizer spreaders), too. Through internal sensors in various aggregates, it is possible to record their states and measure properties such as engine loads, PTOs, threshing or chopping drums et cetera. This internal telemetry provides insight into the machine's work and performance. While some information can be directly read from the telemetry data, for example, status messages, error codes, other information is covert in the sensor readings of a machine and must be uncovered through data analysis.

Since many machines are also equipped with GNSS sensors, telemetry data is usually annotated with global positioning information, too. This geo-referenced telemetry data is the primary data source to work with towards process optimization and digital decision support.

Telemetry recording is usually paired with a tele-communications system for sending internal machine data towards centralized systems for storage and further processing [202]. Communication towards the central telemetry storage system is usually established via mobile-cellular services, to which self-propelled machines connect directly. In contrast, telemetry-enabled implements usually transmit data by connecting to the tractor through standardized connections (e.g., via ISOBUS [115, 129]), using its communication module as a relay. Since cell coverage often is poor in agricultural environments [141, 142], telemetry data is often collected and send in batches [202].

2.2.2 Precision Farming Applications

The most established form of digital support systems in the agricultural domain is currently so-called Precision Farming applications. According to Pierce and Nowak, "Precision Farming can be defined as the application of technologies and principles to manage spatial and temporal variability associated with all aspects of agricultural production to improve crop performance and environmental quality" [165].

Following this definition, Precision Farming can be understood as the precise management and conduct of in-field activities by introducing spatial and temporal variability during operations planning and execution. *Spatial variability* means that instead of treating the entire area of a field identical, sub-areas are addressed to their site-specific needs. *Temporal variability* means that successive operations of the same type and on the same sub-area may vary dependent on the specific needs of the crops at the particular point in time, e.g., following the crop's growth cycle.

Precision Farming is both a managerial approach for planning and orchestrating operations for site-specific agricultural work, as well as a set of technologies that enable and assist farmers in executing the work. Its principles can be applied in many agricultural processes, ranging from the site-specific treatments of arable lands and targeted crop cultivation measures to the precise orchestration of crop harvesting. For a recent and exhaustive review of applications, see [163].

2.2.2.1 Precision Farming Approach – Data Analysis and Operations Planning

When viewed as a management practice, Precision Farming can be characterized as a systematic method of improving the decision-making about crop cultivation and harvesting operations under economic and ecological considerations. It is based on a continuous cycle of data collection and evaluation steps conducted before, during, and after executing one or multiple agricultural operations. It is therefore an inherently data-driven approach and very much in line with the ideas around *information-intensive* agriculture put forth by Fountas et al. [97], cf. Section 2.1.1.

Planning and controlling Precision Farming applications requires an information basis that describes the resources, constraints, and operational methods of the agricultural production business. While the farmer can assert information about workforce and machine resources, gathering detailed and site-specific information about the arable land requires technological assistance. Several approaches for sensing and mapping agricultural environments exist, as discussed below. For Precision Farming, common features of interest in agricultural environment maps include: a field's *geometric layout*, its *soil conditions*, the *crop status* during cultivation and the expected *yield* to be gained by harvesting. To properly time operations, recent data is necessary, such that environment mapping must be conducted cyclically or continuously.

Once the required data is gathered, an operation can be planned. This decision-making process entails integrating all the information provided by the different mapped properties into an operational decision. To determine the demand for a site-specific crop cultivation procedure, for example, the distribution of fertilizer, several factors play a role: the status and growth of the crop, the yield response functions that describe the growth behavior of crops given specific input types and rates, the soil conditions, the weather forecast. Ideally, all these factors have to be considered to calculate the inputs applied during an operation, e.g., determining input rates for particular area units during fertilization.

The decider must also account for the various economic constraints at hand and align with the agricultural business's various strategic and tactical goals. Even though the key characteristic of Precision Farming seems to be site-specific treatments that account for spatial and temporal variability, maximizing yield per area unit is not the most relevant criterion during decision making. It is the *economic viability* of the site-specific treatment. The objective is to decrease marginal costs or maximize marginal gains rather than aim for a homogeneous yield distribution based on a heterogeneous field. Opting for conventional practices, i.e., homogeneously treating a field's area, can therefore also be a viable decision, especially given the high investment costs for suitable equipment [162].

In summary, data-driven Precision Farming exhibits all the inherent complexity of agriculture and challenges regarding the integration of different knowledge domains. As a result of this, all Precision Farming approaches are inherently approximative and only feasible with technological support. To enable Precision Farming in agricultural practice, capable digital assistance systems are required. Such systems must not only aid in managing and visualizing the data required for planning but also actively participate in deriving strategies for site-specific treatments under the constraint of economic viability [162]. Currently, Farm Management Information System and Fleet Management System are the digital tools that may support farmer's in planning precision operations. They will be discussed in Section 2.2.4 and Section 2.2.5, respectively.

2.2.2.2 Precision Farming Technologies – Operations Execution

To conduct Precision Farming operations, a set of technological tools have been established. Currently, tractors with task-specific implements are the most common setup used to carry out Precision Farming tasks, and the usage of GNSS for localization and ECU-controlled implements for site-specific and variable rate treatment is shared among the various established methods [15].

Variable-Rate Applications Variable Rate Application (VRA) aims to enable site-specific adaption during an agricultural operation. In contrast to conventional farming, where traditionally the same amount of input per unit area is distributed uniformly across the field, VRA address its heterogeneity and vary the input per area unit depending on site-specific demands. This can mean increasing or decreasing the quantity of input that goes into the cultivation of crops. When viewed from an economic perspective, VRAs are applied to optimize the marginal costs of cultivation by increasing yield output or saving input resources. In both cases, site-specific applications are less cost-effective in inhomogeneous fields due to increased planning efforts and required technical equipment.

VRA have been applied in a range of cultivation operations ranging from fertilizer, lime, manure, pesticides, but also planting, seeding, weeding and irrigation [163].

Automatic Guidance Systems Automatic Guidance Systems (AGSs) are onboard driving assistance systems that provide automated steering for agricultural machinery.

They are technically based on GNSS sensing, and Electronic Control Units (ECUs) controlling the machines' drive train and steering systems [163]. Standard features include steering along straight or curved lines. Commercially available AGSs are generally focused on steering agricultural machines during in-field operations and do not address steering outside of the field. AGSs can be used to automatically guide a machine along field tracks, execute turns or combine both to follow predetermined sequences of track and turn segments covering the field. The path that is to be followed has to be provided as input for the AGS and can either be set manually by the operator or is algorithmically constructed, as discussed in Section 2.2.5.

Furthermore, distinctions must be made between systems that require the machine operator to regulate the vehicle's velocity and operate the breaks and those that provide automated velocity control [198]. If based on very accurate RTK-GNSS receivers, AGSs operate within an accuracy of of a few centimeters in reference to the target path [101]. For AGS systems that provide automatic turn execution, several geometric models have been proposed to calculate trajectories for different kinds of headland turns [38, 122, 219]. There are additional mechanical or optical guidance-based solutions available for steering harvesters, by detecting the crop row's edge and steering the harvester locally along this line [65].

Using an AGS can significantly improve the precision with which a field is traversed, resulting in a more efficient spatial coverage and minimized spatial overlap of neighboring field tracks during operations [188]. AGS have been shown to reduce fuel and input costs, while increasing work speed and driver comfort [198]. They also reduce soil compaction and prevent driver's fatigue during long work days [17]. AGSs are among the more adopted PF technologies and have been used for field operations such as seeding, tillage, planting, weeding, and harvesting [138]. They can be used without additional VRA equipment but obviously enhance VRA operations because the prescribed applications can be executed as planned. If multiple VRA operations are executed, AGSs can be used to guide operations along the paths of preceding ones [198].

In combination, VRA and AGS systems comprise the most established technical solutions for Precision Farming currently available in the agricultural market. Analyzing their common properties, one can find that both system types implement control loops *onboard* of the agricultural machines. As such, they are designed to function *independent* of additional input during execution. While it is common today that Precision Farming tasks are planned and prepared in an FMIS and then transferred wireless onto the machines, it is not yet established to include an online evaluation of the process's progress or add decision assistance during execution.

2.2.2.3 Advantages, Challenges & Future Trends of Precision Farming

In summary, the Precision Farming approach and a multitude of enabling technologies were developed as tools to increase agricultural production systems' profitability. Likewise, they can reduce the environmental impact of agriculture by preventing excessive inputs and enhancing the ecological sustainability of arable land, e.g., by preventing soil damage. Additionally, Precision Farming approaches can support the practitioners to properly comply with regulatory demands and increase the traceability of agricultural goods, since a detailed pre-operational planning and post-operational documentation of activities is intrinsic to the method [17, 200].

Due to these benefits, the potential to use precision farming approaches in Europe, as well as worldwide, has been estimated to be very significant [205, 234]. However, since the economic viability of precision farming is very dependent on farm size and field conditions and initial investment in specialized technical equipment high, the overall adoption is slow.

Currently, it is considered to be only economically feasible for medium to large farms. Pedersen et al. surmised that “despite all these developments, there is still a lack of adoption among farmers, especially small farms, and the economic benefits from variable rate technology still appear to be inconsequential” [162].

It is, however, expected that over time the required initial investments will decrease. Future trends, such as an increase of real-time mapping, as well as increased automation and the introduction of agricultural robotics, are also already under development. An increase in related decision support systems are also requested and expected. Within their review, Pedersen et al., for example, demanded that “companies and researchers should continue the development of scientifically sound decision-support systems based on real-time information, soil monitoring, weather forecasts and the field history” [162].

2.2.2.4 Precision Harvesting

Regarding agricultural logistics, the subject of precision farming is relevant for two reasons:

Firstly, the principles of the precision farming approach can be applied beyond agricultural cultivation. Using data-driven analysis and digitally-aided planning to figure out the most effective and economical way of working with the agricultural production system is generic, and can be applied for harvesting processes and agricultural logistics, too. When precision *cultivation* is about finding and applying the minimal effective “dose” to stimulate the bio-physical system, then precision *harvesting* and precision *logistics* are about determining the minimal effective “work” to be exerted by the operating system to treat the bio-physical system.

Secondly, the technology that currently enables precision cultivation will enable future precision harvesting and logistics. As our review has shown, the digitization of agricultural machinery has matured to a point where digital control loops, including continuous sensing and (partially) autonomous acting, are possible. Today modern agricultural machines are fully enabled to produce and consume data, which allows further research on how agriculture is currently done and how it can be improved.

Automated Loading & Platooning For example, automated loading systems provide automation for moving a harvester’s loading boom to fill a transport vehicle’s trailer. Currently, commercially available solutions are based on camera-based sensing and classical computer vision approaches [65, 80]. In the context of academic research, model-based approaches are under discussion, too. For example, Happich et al. propose automating loading sequences through analyzing the GNSS positions of both machines and modeling a trailer’s fill level by estimating the pouring cone position and pouring rates [112].

Similarly, Automated Platooning System (APS) have been proposed to provide automated steering for multiple machines during cooperative work [222]. These systems will be built upon AGSs and provide adaptive speed control and steering for the transport vehicle to follow the harvester along an optimized path for loading. Since safety is an essential factor in these systems, latency-free machine-to-machine communication is highly relevant and currently subject of ongoing research [123]. Hence, APSs are not yet commercially available.

Controlled Traffic Farming Another relevant practice in the context of precision harvesting is Controlled Traffic Farming (CTF) systems. These systems "are based on the principle that all the traffic inside the field is restricted to specific tracks only" [198].

CTF approaches were first introduced to reduce the severe soil compaction caused by heavy agricultural machinery and tractors by limiting the affected area to a minimum [132], but have been shown to improve general field efficiency and fuel consumption, too [34]. Route planning algorithms for CTF system with optimal field coverage exit, cf. Section 2.2.5.

2.2.3 Agricultural Environment Mapping & Maps

Maps are an essential prerequisite for digital agriculture in generals and for Precision Farming approaches in particular. The requirements for environment maps are dependent on the agricultural operation, and the dimensions of interest and the required frequency of map updates vary with the given use cases. Likewise, how the mapping of agricultural environments is conducted differs, and suitable sensor systems must be selected and combined with an adequate geo-positioning system to enable a geo-referenced mapping of arable lands in the desired format. Common procedures include *remote sensing* and *ground-based sensing* approaches.

In remote sensing, satellite or aerial pictures are used for mapping. The deployed sensing systems include radar, hyper-spectral, or thermal imaging [17]. Remote sensing data is usually obtained from commercial vendors or provided by governmental agencies, making the recorded data available as a commercial or open service since individualized remote sensing is too costly. Remote sensing systems provide regular map updates, though the spatial coverage around the globe may vary, as does the exact frequency of map updates. Both are dependent on the system deployment, e.g., satellite orbits. A hindering factor in remote sensing is occlusions in the images due to clouds. These may severely affect the usability of remote sensing data.

In ground-based sensing, sensors are either attached to and moved by GNSS-enabled agricultural machines or deployed *in-situ*, i.e., in or next to the field, to directly record and transmit data from a particular location. This can be advantageous because data can be read immediately into the agricultural assistance systems. Commonly used sensing systems include RGB-cameras, LiDAR sensors, multi- or hyper-spectral cameras, too. Recently, the use of Unmanned Aerial Vehicles (UAVs) became a subject of increased research and development, as it promises to combine the benefits of remote sensing with the benefits of more localized sensing approaches [17].

Geometric Mapping & Maps Geometric maps are concerned with the spatial features of agricultural environments and can be mapped in *2D* or *3D* space.

The most prominent 2D features of interest are the spatial boundaries of fields, as well as relevant Point of Interests (PoIs) and Region of Interests (RoIs) related to them. The latter may include annotations describing the location of field entry points or in-field obstacles and areas that represent a field's sub-division into headland and inner field areas. This information is relevant input for planning cultivation and harvesting operations. Similarly, mapping driven routes, either as in-field tracks or street routes, is relevant in terms of 2D lines, too, as it provides input for route planning and automatic guidance systems.

Basic 2D information about field boundaries can be extracted from dedicated land registries, which are maintained by governmental surveying agencies, or manually provisioned using 2D annotation tools [17]. Here commercial and open-source map providers, such as Google Maps or Open Street Map [110] are a valuable input to FMISs, which commonly combine such maps with annotation tools and a backend Geographic Information System (GIS).

Besides using third-party sources, several agricultural PoIs and RoIs can be extracted through analyzing telemetry data, too. In [137], Lauer discusses how to calculate field boundaries, reference lines, and field entry points by analyzing historical telemetry data of agricultural machines. Likewise, he demonstrates how to derive route information and entire road networks from telemetry, too.

Crop Mapping & Maps Crop mapping is concerned with mapping the status and growth of a crop during its cultivation. Crop maps can be derived from sensors mounted on tractors, UAV or through remote sensing [163]. According to van Evert et al., all approaches have in common that "the spectral reflectance of the crop is made into a vegetation index and used as an indicator of the greenness and amount of crop biomass in the field" [162, 218].

Especially, Normalized Difference Vegetation Index (NDVI) measurements are a commonly used indicator to determine current crop status and growth [17]. In combination with additional data regarding the historical weather records and forecasts, crop growth models are used to project the future growth rates and expected yields. Based on this information, site-specific application maps and yield prognosis maps can be created [61]. Likewise, an optimal date for harvesting a particular field can be estimated based on the crop's current and projected ripeness [17, 233].

In-situ crop mapping can also be utilized to immediately control VRAs. 3D geometric crop mapping is becoming increasingly important, too. For example, Redenius et al. discussed how to map the yield in the immediate fore-field of a combine harvester and how such measurements can optimize the control harvesting operations by prospectively adapting the driving speed and motor load depending on the expected material intake [173].

Yield Mapping & Maps Yield maps are spatial maps denoting the yield per area for an agricultural field and allow measuring the total yield amount and analyzing the stochastic yield distribution. Most commonly, such maps document the yield *as harvested*, but yield prognosis maps are also of interest as input to pre-planning precision cultivation and harvesting operations.

Yield mapping targets to map the yield collected during a harvesting operation. Yield maps are constructed using a yield meter in combination with onboard geo-referencing based on GNSS sensors. Yield meters are the internal sensors within a harvester that measure the amount of yield harvested per time unit [175]. It is also possible that additional yield-specific properties are recorded. For example, some Self-propelled Forage Harvesters have a sensor to measure the dry matter content. Depending on the measurement principle, a harvester's yield meter must be regularly calibrated during operations, and the resulting yield measurements and maps are normalized accordingly.

Yield measurements are an essential part of a harvester's telemetry and inform the entire precision farming cycle. Yield maps are usually constructed in data post-processing after the harvesting operation. However, online yield mapping is also relevant because it enables monitoring the current yield performance to improve operational decision-making accordingly. Likewise, consecutive yield mapping over several years can be input for dedicated analysis of the yield potential of individual fields, local regions, or even broader areas. Historic yield data is also used as input for yield prediction.

Yield mapping is closely related to VRA, too. On the one hand, historical yield maps are a vital input since they show the spatial differences in a heterogeneous field, which can be addressed with site-specific application rates. On the other hand, they can be used for post-operational evaluation when assessing whether the precision farming operation was successful or not. In either case: To obtain reliable information, yield mapping has to be done and aggregated over multiple years since the yield distribution can heavily fluctuate on a seasonal basis due to weather influences. Thus a continuous evaluation over the years is advisable. [32]

Of course, there are plenty of other types of maps and mapping approaches relevant to digital agriculture. 3D spatial maps, especially *elevation maps* that provide height information are of interest, describe relevant features, such as slopes, pits, high obstacles, that are helpful to control machines adaptively to their environment. Furthermore, maps regarding soil conditions, groundwater levels, climate, and weather influences can be deemed valuable in agricultural decision support systems. However, since our work relies on 2D maps alone, we omit a discussion of these topics since they are inconsequential for the remainder of this thesis.

2.2.4 Farm Management Information Systems

Farm Management Information Systems (FMISs) are the next category of digital assistance systems to be discussed in our review of digital assistance solution for agricultural logistics.

Within the context of digital agriculture, these systems are regarded as the primary assistance tools for farmers when managing their business. Over the past years, these systems evolved from simple EDP-adided record keeping systems [44] to basic telemetry storage systems [139] into complex systems designed to aid farmers in all aspects of their work [95, 194].

Today a wide-spread range of FMIS products are available on the market and are adopted well by the agricultural community [148]. In 2015, Fountas et al. published a study that reviewed an international of over 140 different FMIS solutions, including marketed products, as well as academic prototypes [95]. The review found that the range of provided functionality is wide and not consistent among all offered systems, yet certain trends can be identified.

In general, FMISs are designed to provide assistance for all aspects of managing an agricultural production business. According to Sørensen et al. an FMIS helps in collecting, processing, storing, and disseminating data in all the ways needed to carry out a farm's operations and functions [194].

It is, therefore, the central system for planning, controlling, and operating an agricultural production system and provides the underlying data management for all data-driven and information-intensive functions and features [97]. In this sense, FMISs combine characteristics of Information Support System and Decision Support System systems, and focus on the farmer as the central decider and provides him with the background knowledge and assistance to aid in the decision processes across all strategic, tactical and operational decisions [195] .

Regarding the conduct of agricultural work, the provided features revolve around planning, monitoring and evaluation agricultural procedures. This entails data management and inventory management regarding fields and machine. Assistance features for planning tasks, e.g., calculating production goals and task planning, are also common. For managing field operations, there's currently a strong focus on preparing VRAs using tractor-implement systems. Depending on the product's focus, FMISs are also used for managing life stock.

To make resource planning easier, task management features often directly connect to classical ERP functions, e.g., human resource planning and work day scheduling. Similarly, finance applications are also common within FMISs, to aid in budgeting, accounting and invoicing, as well as calculating operational cost or total costs of ownership.

They also address the B2B communication and collaboration to connect with business partners, agricultural contractors and other service providers. Likewise, the systems directly connect to digital agricultural machines and the telemetry systems provided by machinery manufacturers. To aid in quality assurance and regulatory compliance, data transfer to governmental agencies or other organizations of the broader agri-food production chain is also in scope [193].

2.2.4.1 Functional Architecture & Technical Implementations

Most available FMIS solutions are based on a centralized architecture [95, 194], because they need to integrate with various sub-systems and external systems, to provide their functionality.

Most available products range from stand-alone or web-enabled desktop applications [95] to web-based services with accompanying mobile apps for smart phones and tablet [151] using proprietary and strongly centralized backend systems. Recently, cloud-based deployments of FMISs and related services have become of interest [121]. Concepts regarding open service platforms and highly distributed service architectures are also proposed [23].

2.2.4.2 Challenges and Future Developments

Though FMISs have become standard tools and numerous products are available to the agricultural community, there are still open topics.

A common criticism is that the initial effort for setting up an FMIS is too high and provides little benefit due to the lack of customization or rigid constraints in the provided products [148, 194], and that current solutions are not yet intuitive enough for the farmer's daily work [95].

According to Fountas et al., there is still a gap in seamlessly integrating the the user's in-depth knowledge about his own company's capabilities and constraints in conjunction with the vast amount of agricultural data collected within the systems. Based on their review, the authors provided an extensive outlook on future features demanded by the agricultural practitioners.

In [95], they emphasize that "the basis for enhanced decision making is availability of timely, high-quality data" and stress that the "increasing needs of farmers and agricultural advisers for time-critical, up-to-date, and precise information as part of farm management" are not matched by existing FMIS solutions. They criticize that "most data and information sources are fragmented, dispersed, difficult, and time-consuming to use" and thus the "full potential of such data and information are not being fully exploited".

To address these issues, the authors claim that the analysis of historical *spatial and temporal data* must be intensified both and incorporated with suitable knowledge representations to coherent management information system. Likewise, the necessity for providing insight into agricultural work in *real-time* and enabling farmers to manage *ongoing operations* while incorporating pre-planned constraints and related economic considerations is stressed repeatedly.

Due to the complex interdependencies that exist in the agricultural domain, the authors also highlight the necessity of proper knowledge management, that is capable to relate between the different facets of managing an agricultural business. This illustrated in the following quote:

> A crucial aspect of FMIS is the knowledge management within the decision processes in the form of dedicated DSS. The development of knowledge-based system in the farming sector requires key components, supported by Internet of things, data acquisition systems, machine-to-machine communications, effective management of geospatial and temporal data, traceability systems along the supply chain, and ICT-supported stakeholder collaboration. The process of building knowledge-based systems for agriculture will be supported and supplemented by industrial developments [139]. Special attention should also be given to interoperability and the availability of standardized formats used on defined data infrastructure elements in the agrifood sector, advanced by, organizations such as the Open Geospatial Consortium (OGC). [95]

We fully subscribe to the above assessment and partially motivate our work on these concerns. Our spatio-semantic approach to reasoning about spatial relations in the agricultural domain, and extracting process knowledge through this process, is very much in line with these observations. Up to now the inherent complexities of agriculture have not been solved and a lack of formal knowledge representations hinders automated knowledge discovery and reasoning across domains. We, therefore, choose to use semantic models to describe the agricultural process, such that automatic inference becomes possible, and also used OGC-compatible spatial representations to contribute to interoperability of spatial environment data, too.

2.2.5 Fleet Management Systems

Fleet Management Systems (FMSs) are another important category of digital assistance system in the agricultural domain. They are the tools that support agricultural practitioners to conduct agricultural operations that are executed by fleets of machines rather than by a single machine. The core functions of an FMS, therefore, address the planning, controlling, and evaluating of cooperative agricultural processes, such as harvesting operations [196].

Based on the above definition, providing support for managing field operations and agricultural fleets can be understood as key functions of FMIS, and it is valid to subsume FMSs as sub-systems of FMISs. However, FMSs can also be regarded as stand-alone tools for agricultural contractors, who provide their machinery and labor as a service to farmers. Managing fleet operations is their core business, and from their perspective, FMSs can be considered as the most relevant category of digital assistance systems. Much like FMISs, commercially available FMSs solutions for agricultural contractors provide business-related functions regarding finance, accounting, human-resource management and others [96].

For the remainder of this section, we will disregard business-related aspects and focus on the functions and technical solutions revolving around the tasks required for fleet management.

We structure our summary on the management tasks for agricultural machinery management, as discussed in the excellent review by Bochtis et al. [43]. These are capacity planning and task times planning, scheduling and route planning, and performance evaluation, cf. Section 2.1.3.2.

2.2.5.1 Capacity Planning

Capacity planning is about determining the set of resources applied in an agricultural operation.

In [43], Bochtis et al. provided the following definition:

> Capacity planning is part of the system design and concerns both a qualitative and a quantitative selection of production components (i.e., in the particular case, machinery and supporting equipment) as related to the demand. [...] The objective is a generic optimisation of the use of the components. [...] Capacity planning is governed by 1) demands of the operation to be performed, 2) availability of equipment, 3) possible working methods, 4) dimensions and capacity, and 5) cost, in addition to the consideration of functions such as available labour, timeliness, and workability. [43]

When viewed as a strategic decision, it is about allocating resources for the entire agricultural business, e.g., buying new machines. Søgaard and Sørensen proposed a model for determining the optimal level of farm mechanization [191]. Their method accounts for the demand of agricultural work, the machines' technical capabilities, and potential utilization rates and estimates the total costs of ownership for potential machine acquisitions. It considers constraints such as available person-hours and time windows with peak workloads during the year.

Similarly, Amiama et al. used simulations to study the effects of different fleet setups of harvesters and transport vehicles on the accumulated costs of an entire season of the harvesting [12].

When treated as a tactical decision, capacity planning is about determining the resources of a single harvesting operation. Here one can assume that the particular demand and available resources are known. So the planning problem poses the question: Which resources shall be allocated to form the harvesting fleets for this particular set of fields?

Busato et al. addressed this question for large-scale silage production operations [55]. Their work aims to optimize the entire logistics chain by adequately balancing the capacity of transport vehicles, with the production rates of the harvesters and the storage rate of the compactor vehicles. Their approach combined event-discrete simulations with a linear programming-based optimization model. At first, multiple simulation runs were used to determine the "total operating time and total operation cost for different configuration of the allocated transport units based on machinery and field features".

Their simulation accounted for field features (area, distance from the facilities, yield, shape) and machine features (production and storage rates) and varied the number of transport vehicles and the transport rates. However, all transport vehicles were assumed to be of the same capacity and running with the same transport rates within a simulation run. Based on these results, the cost matrices for the optimization were constructed. The number of used transport vehicles was used to "minimize the total operational cost of the production system under the presence of time constraints for the completion of the operation." Based on their experimental results, analyses regarding the utilization rates of production, transport, and storage while accounting for total costs were presented.

A similar approach was provided by Steckel [201]. In his dissertation thesis, he worked on the very same problem of analysis resource allocations for harvesting operations. His method also applies event-discrete simulations and provides a set of equations with which the total cost for the individual process segments and vehicles can be calculated directly. The presented analyses transparently show the theoretical limits of performances and costs to be achieved without running simulations at all.

Both works provide an in-depth analysis of how different sets of resources allocated to the same field condition may behave during operations. Such analysis provides valuable insight into the trade-offs that come with adding or removing transport vehicles to a harvesting fleet and, thus, can support practitioners to configure their harvesting fleets better.

2.2.5.2 Task Time & Field Order Planning

Task time planning is about estimating the required time duration of activities. It is closely related to scheduling but more focused on determining suitable time frames to conduct certain operations and bringing operations in a suitable order [43].

To support task time planning, Martin-Clouaire and Rellier proposed a simulation-based approach for conflict-free allocation of resources for a set of agricultural tasks [143, 144]. Bochtis et al. worked on planning multiple operations in a row by using flow-shop modelling [33, 40]. In [33], they presented how to order operations across several geographically dispersed fields where biomass handling operations have to be carried out. The original approach only accounted for planning a single machine's operation, but in subsequent work, Orfanou et al. extended it for multiple vehicles [156]. This work also accounted for operational costs in detail.

Basnet et al. presented a model for planning the succession of multiple harvesting campaigns as conducted by contractors that service multiple harvesting operation [20]. Their model accounts for lag times between operations, when resources have to move from farm to farm, but did not consider the readiness of fields. This aspect was addressed by de Wall et al., who analyzed crop ripeness using remote sensing data to optimally date the harvest operation begin and as a criterion for scheduling a harvest's day-to-day operations [76].

Such approaches can effectively support practitioners in planning harvesting sequences and determining plausible task assignments for harvesting campaigns.

2.2.5.3 Scheduling, Area Coverage & Route Planning

Once resources are allocated and rough task times are defined, the particular activities of all involved machines can be scheduled. Scheduling is about breaking down timetables into actionable sequences of distinct tasks and, thus, a refinement of the previously made plans. Since operations are usually spatially dispersed, scheduling commonly incorporates some form of spatial planning related to field operations [43].

Most commonly, spatial configuration planning techniques create a geometrical representation of a field's area, which can be used for subsequent planning. Three aspects are of special interest:

the division of a field area into useful sub-areas, determining driving direction within each sub-area, and constructing field tracks that fully cover each sub-area.

There are many contributions to spatial planning for agricultural fields. Palmer et al. discuss the importance of guidance lines when creating in-field tracks [159]. Furthermore, Hameed et al. generated automated guidance lines by analyzing telemetry [111]. Scheuren proposed methods for creating in-field tracks for full-field coverage along guidance lines [180, 181]. Spekken & de Bruin focused on reducing non-productive work by minimizing maneuvering operations during track generation [199]. Bochtis et al. proposed additional techniques for minimizing risk of soil compaction [42] and planning of tracks for Controlled Traffic Farming operations [34].

Based on these spatial representations of field tracks, field entry points, and other geometric primitives, one can plan for field area coverage and machine-motion sequences [39].

An area coverage plan determines how one or more agricultural machines have to traverse all points in a field under criteria such as minimizing cost, time, and overlap [39]. It thus determines the overall strategy for harvesting a field. Machine-motion sequence generation, also called (in-field) route planning, concerns the optimal connection of the field tracks provided by the spatial configuration plan. It is about finding the optimal sequencing of the field tracks and the optimal sequence of working the sub-fields.

Since these track sequences have to be assigned to one or more agricultural vehicles, the route planning problem is best solved in conjunction with the scheduling problem.

For agricultural operations, Bochtis and Soerensen addressed combined route planning and scheduling can be solved simultaneously by utilizing combinatorial optimization techniques [36, 37]. Their approach builds on the concept of Vehicle Routing Problems (VRPs) [215]. This family of optimization problems was formulated to generalize the traveling salesman problem and concerns scheduling a fleet of vehicles that are supposed to visit each of a set of customers exactly once and deal with the customer's demand. Formally, a VRP is modeled as a weighted graph for which an optimal traversal is to be found. The graph is constructed by a set of nodes and edges, where the node-set consists of the vehicle depots and customer nodes, which have to be visited by one or more vehicles. The edges between the nodes represent the costs associated with traveling between the nodes. Solving a VRP aims to find an optimal traversal of this weighted graph, such that each customer's demand is matched while the costs are minimized. The concept of a VRPs has been developed into a family of optimization problems depending on specific criteria and constraints of the application to be solved, e.g., to account for temporal and spatial constraints during traversal and capacity constraints of the vehicles, as well as deterministic, stochastic, or unknown distributions of customer demands.

Initially, VRPs were developed for the scheduling of classical supply chain management problems and the routing of industrial goods [215], VRPs have also been successfully applied to other real-world fleet management problems. Applications range from scheduling of public transport to urban waste collection to managing customer visits for services and sales people [215].

Bochtis and Soerensen first described how VRP theory is applied onto the agricultural domain in order to schedule field operations [36, 37].

According to their model, work demands are represented as customer nodes. The fleet of agricultural vehicles has to visit these nodes and fulfill the respective demand, i.e., do the agricultural work, such as harvesting an area. The individual tracks of a geometric in-field route planning, for example, can be understood as such customer nodes, and by solving the VRP, an optimal sequence of traversing these tracks can be found and assigned to harvesters and transporters. The graph's edges describe the associated costs, such as the time required work, the distances to drive, the fuel spent, and the yield gained.

Since there are many variations to VRPs, the authors cover the theoretical foundations of how to model various agricultural operations. Distinctions are made for single machines or cooperative machine operations, for the type of material flow (inbound, outbound, none) and different demand characteristics (deterministic, stochastic, unknown). The VRP solutions presented in [36] focus on operations executed by primary units that either work alone or in parallel but independent of each other, e.g., VRA applications. The VRP solutions presented in [37] concerns planning with interdependent machines. Here a fleet of primary units (e.g., harvesters) are collaborating with a fleet of secondary units (e.g., transporters) such that the solution of the VRP assigns unit pairs to meet at the customer nodes. In this case, there are actually two VRP problems to solve - one for the harvester servicing the field and one for the transporters servicing the harvester. Since the progress of the harvester in servicing the field is dependent on the stochastic yield distribution. In this context, the process of forage maize harvesting can be understood as a VRP with time windows with stochastic demands [59].

In general, VRPs are considered NP-hard, but approximative solutions can be found. General methods have been proposed for a-priori planning and real-time optimizations to solve this kind of stochastic optimization problems [166].

Later on Bochtis et al. refined their method for optimizing field coverage [41], minimizing headland work [38] and path planning for service vehicles [35].

Jensen et al. addressed inter-field and intra-field routing in a combined VRP [118]. Their approach focused on planning routes and schedules for transport units servicing combine harvesters. It used optimization criteria, such as time and traveled distance, and generates paths for both in-field and between field movements for the transport units. The plan space is based on pre-planned in-field track geometries and external road network data. Their method further allows to impose rules of controlled traffic farming onto the generated paths and differentiated between rendezvous of harvester and transporter for loading in stand-still and ongoing motion.

Edwards and Jensen address the subjects of field coverage and route planning for machinery with limited capacity [86, 87]. They present a control system that monitors the capacity change and calculates an optimal coverage path in real-time. Their approach is also capable of mapping the spatial variability in the field "to enable an operator to monitor the situation or for use in subsequent operations" [87]. Their approach targets harvesting and slurry operations.

In [120], Jensen et al. recorded and analyzed GNSS traces of transport vehicles during slurry application runs. They segmented the recorded routes into segments by classifying several actionable tasks (such as headland turns, traveling on (un-)worked track and categorized them into productive and non-productive activities. This qualitative analysis was then used for optimized planning. The method presented in [119] extends the approach presented in [87, 118] using state-space search techniques in combination with solving the traveling salesman problem.

Edwards et al. worked on planning optimal task sequences with required delay times between tasks. Their method was applied to plan grass harvesting operations in which mowed grass has to dry before collection. Their method accounts for the readiness of a field for a particular action, based on previous actions, required delay times, and likeliness of rainfall and outputs work plans for the individual machines of a fleet tasked to execute multiple consecutive operations at multiple fields. They present examples for planning activities across multiple days, which renders their method suitable for task time planning and scheduling. An evaluation of the system showed a generally good match when comparison of planned routes versus driven routes [88].

In [185], Seyyedhasani et al. discussed allocation and ordering of field paths among a number of available machines while optimizing the completion time for the entire field. Their approach utilized the application of Tabu search [103] to solve the VRP problems and compared its performance against the commonly used Clarke-Wright algorithm [66]. In [187], the authors validated their approach in real-life grass mowing operations. They recorded the routes driven by human operators during the first mowing operation and create a VRP representation based on this data. For the second mowing, the drivers then followed the routes generated by the optimization procedure. The subsequent analysis showed that “the predictions by the VRP representation for completion time and total time were both within 2% of the actual times recorded when the tractors followed the computer-generated routes in the field”. Based on the optimized paths, “the time to complete the fieldwork was reduced by 17.3% and the total operating time for all tractors was reduced by 11.5” compared to the first mowing operation.

In summary, many promising approaches to solving the various planning problems around managing an agricultural harvesting process and its logistics aspects exist.

Some of the above planning solutions already directly contribute to controlling agricultural operations today. Especially, VRA and CTF operations highly benefit from planning systems for optimized routes [17, 198]. Through AGS systems, these are also easily deployed in practice.

However, most of the above solutions regarding agricultural logistics remain academic studies.

2.2.5.4 Dynamic Planning & Control

Similarly, dynamic planning and plan-based control are subjects of ongoing work.

Sørensen and Bochtis provide fundamental concepts for integrating planning and execution systems [193]. According to their proposal, future FMSs have to be closely connected to the onboard control systems of agricultural machines, such that plans can be monitored and revised to optimize the process despite unforeseen events. They also sketch how planning and control functions can be distributed across various centralized and decentralized system components to form a hybrid control system capable of addressing the uncertainties of agricultural work through top-down and bottom-up approaches.

In [43], Bochtis et al. review the state of the art of agricultural planning systems and propose subjects for future work. They provide an architectural schematic that illustrates how planning and control modules shall be integrated for optimizing ongoing processes, cf. Figure 2.8.

The authors suggest that existing capacity and task time planning solutions must be more adaptive and include data derived from the ongoing process. Likewise, specific planning components for dynamic route planning and real-time coordination of resources and task allocation

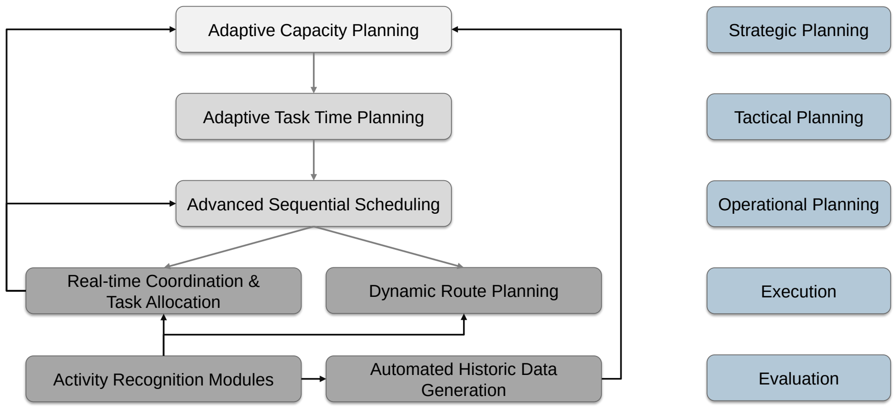


Figure 2.8: An architectural schematic for connecting planning and monitoring modules to facilitate dynamic control for ongoing processes. Reproduced and adapted based on [43].

need to be designed. Their concept also calls for dedicated activity recognition modules. These are intended to evaluate the ongoing process and directly feed into real-time coordination and task allocation and dynamic route planning systems. Furthermore, a module for automated history generation is proposed, providing input for future strategic and tactical planning.

Regarding the subject of dynamic route planning, Seyyedhasani & Dvorak worked on dynamic re-routing of vehicles and reallocation of machine resources for covering an agricultural field. In [186], they proposed a dynamic VRP model with multi-depots, which can represent the current progress of the fieldwork and allows machines to pick up work on partially finished in-field tracks. It is used for optimizing vehicles routes for field efficiency and is adaptive to specific progress metrics accounted for during execution.

The approach was tested in simulations that would vary the number of available transport vehicles and alter their performance in terms of vehicle speed, hence, resulting in varying area performances during the simulation runs. Re-plannings were triggered at fixed time points during each simulation at 25%, 50%, and 75% progress. New plans were generated if either the progress made per in-field tracks was below or above the expectations of the previously made plan, or if machines could no longer be allocated, e.g., due to a *virtual* machine failure.

The authors subsequently compared the different simulations and analyzed the impact of re-planning on the overall field efficiency. They observed that re-planning can be advantageous, but that the significance of deviations from the original plan has to be accounted for:

> The results revealed the impact of the new routes is dependent on the specifics of the event that necessitated the re-routing. When a vehicle was added to the fleet working the field, the updating procedure was able to use that vehicle to reduce completion times. When vehicles operate at unexpectedly fast or slow rates, recalculating the routes can improve field work parameters, but only if the change in work rates caused a significant deviation in field work progress. – [186]

Hence, updating a plan if the execution is only slightly off the expected performance metrics can also disturb the process and lead to inefficiencies.

Towards connecting multiple planning systems, de Wall et al. put forth concepts on how capacity planning and scheduling can be connected to execution monitoring systems to enable dynamic planning [76]. Stock et al. propose to use techniques of solving constraint satisfaction problems with meta-constraints for connecting multiple harvesting (sub-)problems to enable hybrid planning [203].

Working towards increased decentralization in planning and control systems, Quindt et al. discuss how agent-based systems could be used to decentralize decision-making in connection with a harvest planning system [172]. In the same line of work, Kremer and Westerkamp sketch how an onboard decision support system could aid transport drivers during process execution by displaying telemetry data and task information [131].

More directed towards developing fully autonomous agricultural machines, Scheuren et al. demonstrate how robotic techniques and optimized in-field route planning can be combined to let agricultural machines autonomously harvest an entire field [182].

The above solutions show that the importance of plan-based control is understood and that the existing systems for harvest-related planning can be adapted towards more dynamic solutions. However, the review also reveals that adaptive control and dynamic planning is still an open subject in the agricultural domain. It is also apparent that even though the existing works on dynamic control notice the importance of monitoring the ongoing process, they do not address this subject adequately. This is a short-coming, because without the proper monitoring solutions or activity recognition modules, the input for real-time coordination and planning is missing.

2.2.5.5 Process Monitoring & Evaluation

Solutions for monitoring agricultural processes concern evaluating ongoing or past operations and providing the necessary feedback for the related decision-making, management, and control.

Bochtis et al. provided the following definition:

> Machinery performance evaluation regards the final step in planning and control cycle for a field operation. A key point is the comparison between the planned operation and the actual executed operation. The result of this comparison has to be integrated in the subsequent repeated planning cycle and will enable the manager to adapt to the operations planning process. – [43]

Additionally, Sørensen and Bochtis summarized that the evaluation of a field operation has four different goals: data processing for documentation, documentation for compliance, analysis of operational performance, and comparison with production targets and operational goals [196].

Some works have addressed the monitoring of agricultural processes, mainly as off-line solutions that evaluate operations after their execution ended.

Sørensen and Nielsen used manual monitoring to classify and measure machine activities during tillage operations [197]. Grisso et al. processed GNSS telemetry to evaluate the difference in vehicle speeds for straight and curved in-field tracks [105]. Their method, however, did not separate activities into different process states. It also involved manual interventions, such that it can be considered to be semi-automatic. Askey semi-automatically segmented historical data sets into different activities and derived quantitative assessments of process times and production rates from relevant process segments [13]. Taylor et al. analyzed process times for harvesting data obtained with a yield monitor and GPS receiver in a full-automatic fashion. The work focused primarily on the field efficiency of harvesters and did not consider transport vehicles [207].

Many of the reviewed works on process planning also incorporated some monitoring-related data processing to provide comparisons of planned and executed operations [88, 88, 119, 187]. However, no details on the particular methods are provided and seem to rely on dissection of the recorded telemetry or other semi-automated analyses.

In [117], Jensen and Bochtis addressed the problem that "the acquisition of data from the operations execution requires a high level of manual invention". They present methods for automatic recognition of operating modes of agricultural machines involved in grain harvesting.

Their work applied agricultural expert knowledge to create pre-determined analysis models that combined spatial analysis and computational reasoning over the recorded telemetry to infer process information. Their approach is based on analyzing raw GNSS trajectories and is capable of capturing the most relevant in-field activities for harvesting operations:

They utilize a coverage map and intersection tests between the area currently covered by the harvester against all previously covered areas for detecting harvesting work. If the harvester is on a new patch of land, their system classifies its activity as active harvesting. For detecting loading sequences, they rely on measuring spatial distances between harvesters and transporters and analyzes the vehicle speed to differentiate between stationary and on-the-go loadings.

They applied their analysis functions on recorded telemetry. This was done in post-processing, but the prototype processed multiple hours of recorded data in a few minutes. Even though their approach is not designed for online monitoring, the proposed methods evaluate process states at individual points in time. This makes them, in principle, applicable for a monitoring system that operates in near-real-time, too. Hence, this work has a significant impact on this thesis and serves as a blueprint for implementing the classification of process states.

In a related report, Jensen also worked on detection methods for transport vehicles queuing to be loaded by a harvester and proposed hierarchical applications of low-level classifiers to differentiate between process states within and outside of the field [116]. The work on classifying the different activities involved in in-field work by Jensen et al. also relates to process monitoring. Even though they applied manual segmentation in [120], the resulting systemization helps in constructing additional classifiers for a more fine-grained activity recognition system.

The commercial sector provides only a few products related to harvest monitoring and supporting agricultural logistics [30]. Some vendors provide fleet overview solutions that share selected telemetry and positional information within a machine fleet [125], to provide an overview of where partner machines are. However, these provide no insight into the process progress or any form of decision support whatsoever.

To the best of our knowledge, there are no dedicated works on monitoring ongoing harvesting processes and no works utilizing semantic technology to reason about detected process states.

2.2.6 Semantic Technologies in Agriculture

Semantic technologies capture the meaning of data by using formal logic and machine-readable representations. As a research field, the subject is centered around describing concepts and facts in a structured way and deriving additional knowledge. It is about knowledge representation and reasoning and, thus, a sub-discipline of artificial intelligence research and computer science.

As a technology stack, the term refers to a set of established knowledge representations, design principles, and exchange standards that can be used to develop knowledge-based systems.

Most of the commonly adopted tools available today are closely related to the semantic web initiative, which wants to augment the data within the world wide web with semantic annotations and, thus, make it machine-readable [28]. The semantic web is based on several standards put forth by the World Wide Web Consortium (W3C) [230]. The most central ones are the Resource Description Framework (RDF) [229] and Web Ontology Language (OWL) [229], which are used to formally represent knowledge. RDF is a metadata model that specifies how to annotate information to (web) resources and formulate factual statements about the relationship of resources. RDF statements follow a simple subject-verb-object structure, such that RDF-compatible databases are often called triple stores. OWL builds upon RDF and is knowledge representation language used to describe semantic models in terms of ontologies. Ontologies are a formal description that denotes the fundamental categories, concepts, and properties relevant to a knowledge domain [106]. OWL is based on Description Logics (DLs), a family of formal knowledge representation languages designed for balancing expressive power and reasoning complexity [16].

Based on the principles of Linked Open Data (LOD), the semantic technology community is encouraged to describe individual knowledge domains in terms of ontologies and share them through web services such that other ontologies can build upon already defined content. By doing so, it becomes possible to automatically exploit relationships between distributed data sets in various formats and from diverse sources and derive new knowledge by connecting different knowledge contexts through automated inference and reasoning.

Semantic technologies are also designed to enable search, information retrieval, and question-answering. To query knowledge bases adhering to semantic web standards, the W3C specified the SPARQL Protocol and RDF Query Language (SPARQL), a SQL-like language for processing RDF data. It is also standard for knowledge bases that are implemented using semantic web technologies to provide inference mechanisms for ontological and rule-based reasoning directly. In this context, the Semantic Web Rule Language (SWRL) is a common standard to specify rules.

In summary, semantic technologies are a collection of algorithms and tools for structuring data and bringing meaning to information. Semantic web technologies additionally adhere to the W3C standards which are designed to simplify the implementation of knowledge bases systems on the Internet, but in other system contexts, too.

2.2.6.1 Semantic Resources for Agriculture

Agriculture is an inherently inter-disciplinary domain that is influenced by many different knowledge contexts. Hence, formal knowledge representations and cross-context reasoning are beneficial to the agricultural domain. This is reflected in the existing works on utilizing semantic resources to describe agricultural knowledge.

Drury et al. provide a review of existing semantic resources for agriculture [84], Regarding the availability of semantic resources, they concluded that “agriculture has a large number of semantic resources that have been developed”.

The following instances of semantic resources in agriculture are most noteworthy.

AGROVOC With AGROVOC the Food and Agriculture Organization of the United Nations curates the largest controlled vocabulary for agriculture [94]. It contains over 39.000 concept descriptions for agriculture and related contexts including food, nutrition, forestry and environmental resources. It is a multilingual thesaurus that helps in formally describing and translating concepts within the agricultural domain. It provides more than 800.000 terms that unambiguously identify individual concept across up to 40 different languages, and makes extensive use of the semantic web standards, relies on many established core ontologies and is openly available.

agroXML / agroRDF agroXML is an XML dialect designed for data exchange in agricultural digital assistance systems [145]. It provides a defined vocabulary for representing and describing farm work that focuses on work processes on the farm, including common supplies like fertilizers, pesticides, and crops. It contains close to 200 concepts and about 150 assignable properties. According to Martini et al., it is intended “to be used within farm management information systems as a file format for documentation purposes but also within web services and interfaces between the farm and external stakeholders as a means to exchange data in a structured, standardized and easy to use way” [145]. It is supposed to complement the ISOBUS standard commonly for data exchange with agricultural machines. To integrate the vocabulary with established semantic web technologies, there’s an RDF-based derivate called agroRDF with the same content [133].

The standard was initially developed within the iGreen project [27, 82] and is now maintained by the german Kuratorium für Technik und Bauwesen in der Landwirtschaft e. V. (KTBL) [134].

2.2.6.2 Applications of Semantic Technologies in Agriculture

In their 2019 review, Drury et al. also surveyed current applications of semantic web technologies in the agricultural domain [84]. According to the authors, some applications have adopted semantic representations and utilize ontological reasoning, yet this is definitely not common within digital assistance systems for agriculture. In summary, the authors assessed that the application of semantic technologies is currently “underutilized” despite the available models.

Their overview points to some decision support applications regarding rice cultivation, pest management, wine-making, or fertilization. See [84] for related references. These works have little overlap with the subject of agricultural logistics, but some other works are directly related.

agriOpenLink The agriOpenLink project worked on semantic models for precision agriculture [227]. According to Tomic et al., the project’s goal was “to establish formal machine-readable semantic models of several agriculture processes and to develop a semantic service-based process platform to support flexible process creation, monitoring and optimization, as well as flexible creation of new processes as new atomic functions and new equipment are made available” [214].

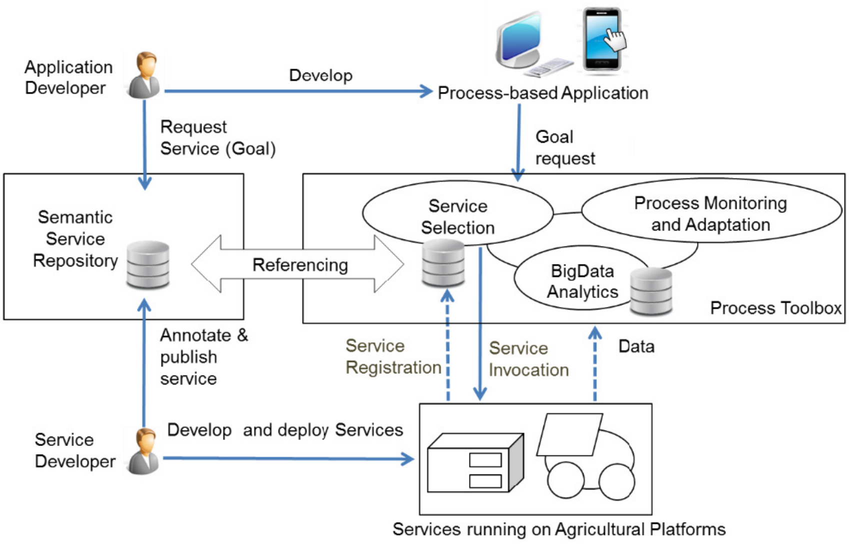


Figure 2.9: A semantic process management architecture as proposed by Tomic et al. [214]. Figure reproduced from there.

In their work, the authors criticized that the process models utilized in commercial PFA solutions are closed source, which makes data exchange between vendors difficult and hinders connecting systems across processes. They also propose that opting for openly shared semantic models for describing agricultural processes and using these as interface descriptions between machines and assistance systems could remedy these problems.

Within the project, Tomic et al. proposed to utilize a semantic process management architecture, cf. Figure 2.9. According to their concept, agricultural processes shall be assisted by a service layer that provides analytics, monitoring, and planning capabilities based on semantic models. This service layer can support both management tasks and onboard systems for guiding work processes through semantic service discovery and querying.

An example application regarding livestock management was developed to test and validate their concept. It includes a domain ontology, queries, and inference rules for dairy farming [213].

Agriculture Operations Task Ontology In [7], Abrahão and Hirakawa conducted a review of several agricultural ontologies with a focus on differentiating existing works into domain ontologies and task ontologies. As discussed by Martin and Falbo, the former provides the vocabulary related to a domain, while the latter captures the details of particular tasks within this domain, usually by extending the domain ontology [146]. A task ontology is dedicated to describing “the knowledge of where and when the tasks might occur, who is responsible for them and what the tasks, attributes and relations are” [7]. The author’s review concluded that there are “no research works about task ontologies to describe agriculture field operations”.

Hence, they worked on ontological models for field operations with a dedicated focus on representing tasks and their actors [7, 8]. Their model is based on an ontology for collaborative tasks in multi-agent systems described by Schmidt et al. [183].

The developed ontology provides basic concepts to describe tasks and their execution. It introduces concepts such as agents that execute tasks, locations at which tasks occur, and resources that are needed as inputs or produced as outputs. It also provides concepts and relations for managing a hierarchy of tasks and sub-tasks and denoting task sequences, parallelism, and other forms of control flow during execution. See Figure 2.10 for a depiction.

The modeling language OntoUML [155] was used to describe the process for harvesting sugar cane using this ontology. Figure 2.11 shows the resulting process description. It structures the tasks of several different types of agents in a set of activities, some of which occur in sequence, some in parallel. It also denotes the input and output resources that go into the process and links to several process-related feature properties, allowing additional data attaching.

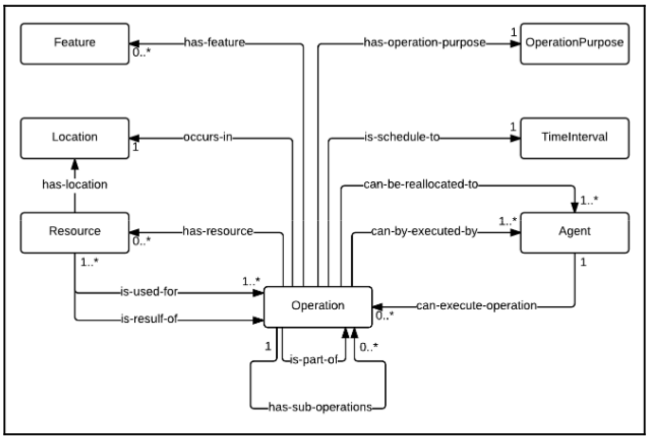


Figure 2.10: The main concepts and relations used by Abrahão and Hirakawa to model agricultural field operations. Reproduced from [7].

The presented work demonstrates how agricultural processes can be successfully captured using semantic models. Unfortunately, the presented work is a pure model, not fit for computational processing, because it is based on OntoUML and UFO [107], which is not immediately compatible to RDF/OWL. However, there are works on automatically translating OntoUML-based models into OWL ontologies plus SWRL rules [19].

In [8], Abrahão and Hirakawa extended the model and named it Agriculture Operations Task Ontology (AGROPTO). The extension included concepts and relations for modeling task objectives, describing interferences with external events, and assigning pre and post-conditions to activities. The model also accounts for different states of task execution, e.g., differentiation between canceled and interrupted executions. See [8] for illustrations of the entire model.

In summary, semantic technologies are recognized to be helpful in agriculture to manage knowledge across different domains, but the benefits have not been fully utilized in application. However, there are several substantial ontological descriptions for describing general agricultural concepts in general and dedicated works on modeling agricultural field operations. These can be reused or used for reference when working towards monitoring agricultural logistic processes in terms of formal knowledge.

Of course, other domains have also adopted semantic technology and often with successful applications. Chatzimichail et al. provide a recent review of semantic technologies in the context of the Internet of Things (IoT) [60]. They point to applications in health care, disaster management, public events security, intelligent transportation, building and infrastructure management, and more.

This thesis will also draw from other domains, especially from logistics, as will be discussed in the included articles.

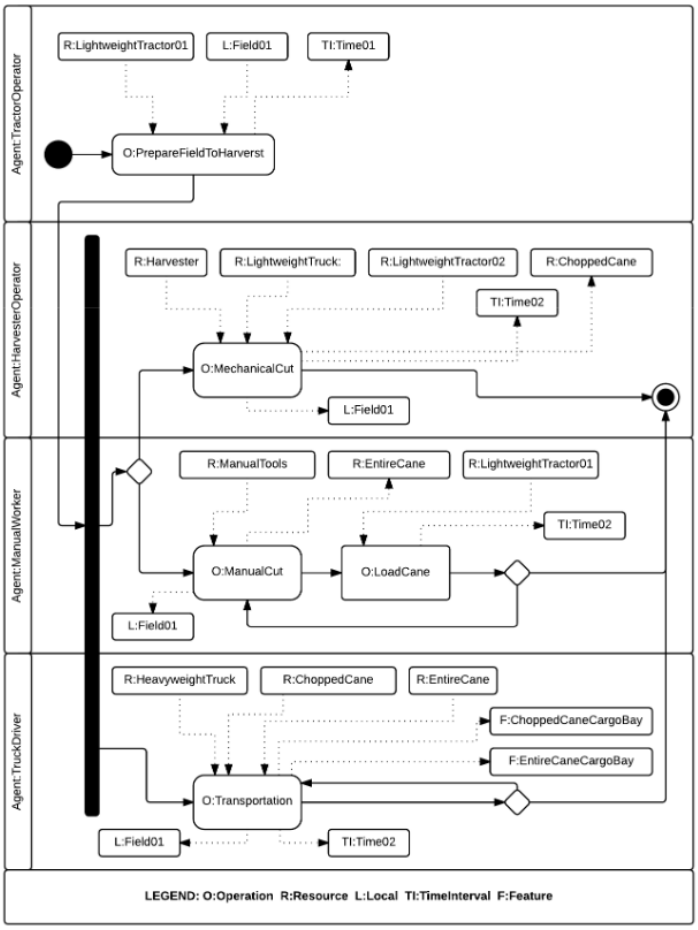


Figure 2.11: A semantic sequence diagram for the process of sugar cane harvesting. It was modeled by Abrahão and Hirakawa using their Agriculture Operations Task Ontology. Reproduced from [7].

2.2.7 Summary & Open Topics

Summarizing our review on digital assistance systems for agriculture, several things stand out:

Planning agricultural operations is very feasible, and there is a broad set of tools to schedule single machines and entire fleets. However, there are open topics regarding dynamic planning, and the integration of planning systems into control solutions is lacking, too. There is also little dedicated work on combining the planning solutions for the different management levels and planning problems into holistic, customer-facing decision support systems.

Digital assistance solutions for harvesting processes are therefore generally limited because these planning problems need to be solved in conjunction and *during* operations. Otherwise, the practitioner has no benefit from it, except maybe for improved initial resource allocation.

Likewise, there is a clear gap in technical solutions concerning the monitoring and evaluation of ongoing processes. Though the necessity of such capabilities is understood and conceptually called for, e.g., in [43, 196], this subject has not been studied in the agricultural domain.

In their 2014 review on fleet management solutions, Bochtis et al. summarized:

> All approaches lack an automated performance evaluation process. The prediction of operating modes for agricultural machines based on automated activity recognition, as observed in other domains, is expected to be a future research topic in the agricultural machinery management domain. Furthermore, all of the current approaches are focused on single machine monitoring. Monitoring of systems of co-operating machines (e.g., harvesters and transport units) is also a future research topic. – [43]

We fully support this assessment and emphasize that it still holds in 2021.

Based on the above review, we also conclude that the necessary pre-requisites for addressing the monitoring of agricultural operations are given. Therefore, the subject can be studied further.

Modern agricultural machines provide a rich set of telemetry that can be analyzed to uncover process-related information and evaluate its progress. Likewise, FMISs and FMSs are established tools in the agricultural domain and can provide the necessary infrastructure for implementing dedicated activity recognition modules. They also provide interfaces relevant to agricultural databases and planning approaches. The creation of dynamic planning systems in the context of these assistance systems only becomes feasible once online process monitoring solutions have been developed. Furthermore, initial concepts for how to implement activity monitoring for harvesting processes have been put forth, which can be extended to account for ongoing operations.

Hence with this thesis, we aim to contribute to the subject of monitoring agricultural operations, especially for ongoing agricultural logistics processes.

The above review also shows that there are currently no proper decision support systems supporting agricultural practitioners in managing the execution of harvesting processes or other agricultural logistic processes. This is because decision support for cooperative processes is a tough problem due to the increased complexity of having multiple machines and multiple decision-makers in a highly dynamic environment.

We assume that working toward analyzing and evaluating ongoing processes can also be beneficial to developing more customer-facing decision support systems in this space.

In [75], Dayde et al. discuss the importance of operational decision making and point out that the current mode of decision making is often experience-based and that agricultural managers are utilizing soft, qualitative, intuitive assessments on how to improve the operation's execution. Through online process analytics, this mode of decision-making can be supplemented through hard, quantitative, evidential feedback. Providing such feedback provides input to the related planning system but should also help practitioners determine whether a particular operational decision was beneficial or decremental to the execution.

Once processes can be monitored in terms of detailed quantitative data, it also becomes feasible to derive qualitative assessments of the current situation. The necessary quality metrics and process targets can be provided through the above planning systems. We believe that working towards qualitative assessments can be used to enable digital assistance systems to explain and argue for proposed actions that should make them more approachable for the user.

Therefore, capturing process information in qualitative and quantitative terms is instrumental in communicating with the user and automated reasoning and control approaches. Finding ways to represent a process and its progress in such a way is, hence, an essential additional research topic that must be considered when supporting ongoing agricultural processes. We believe that semantic representations and automated reasoning will be essential to enabling real-time coordination and task allocation systems, as proposed by Bochtis et al. [43], because they will provide the flexibility in seamlessly connecting systems dedicated to low-level data processing with those tasked with high-level decision making. Based on these considerations, we utilize formal knowledge representations and reasoning in our method.

2.3 Online Process Analytics for Agricultural Logistics

Given the above considerations, the context of digital assistance systems should be extended with an additional system component capable of providing online data analytics for agricultural logistic processes. Developing these capabilities is a prerequisite to further enhancing digital assistance in agriculture towards fully capable decision support systems that use knowledge to reason about the agricultural process. By doing so, it becomes feasible to better support agricultural practitioners in managing agricultural logistic processes based on recommendations and optimizations during execution rather than providing assistance features only before a harvesting campaign is executed or by displaying solely raw telemetry during execution. This thesis aims to propose concepts for this such systems and aims to implement a prototypical system that provides online process analytics for ongoing forage maize harvesting.

In the remainder of this section, we will develop the conceptual basis of online process analytics systems, explain their core analytics functions and sketch their role within the context of digital assistance systems for agriculture. Subsequently, we discuss how the results of the publications included in this thesis contribute to the automatic extraction of high-level process information through telemetry data analysis and spatio-semantic reasoning.

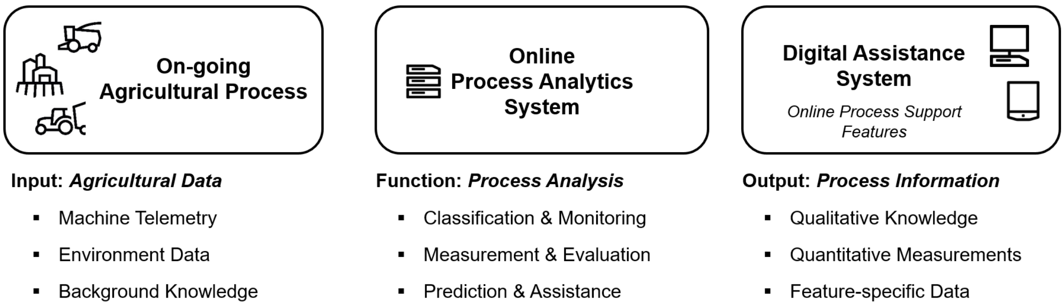


Figure 2.12: An Online Process Analytics (OPA) system receives and analyzes agricultural data, like telemetry, environment data, and process-related background knowledge, in near-real-time to provide qualitative and quantitative knowledge about the ongoing agricultural processes.

2.3.1 Online Process Analytics & Support Features

An Online Process Analytics (OPA) system consumes and analyzes agricultural data in near-real-time and produces qualitative and quantitative knowledge about the ongoing agricultural processes. It provides functions for monitoring and evaluating processes as they unfold and aggregates data or derives knowledge to enable process support during execution.

These functions are realized by analyzing telemetry with respect to a process-specific semantic model. Additional data sources, especially regarding environment data and related background knowledge, may also be included as required by the particular implementation of analytic functions and assistance features. The system may further project into the process's future states through prediction models or evaluate the ongoing processes by matching against provided plans, e.g., by interfacing with a dedicated planning system.

Since the data coming from an OPA system is designed to inform about ongoing processes, it is well-suited to realize any assistances and support features related to process execution. Hence, we call features that provide near-real-time support for the agricultural process during its execution Online Process Support (OPS) features, cf. Figure 2.12. Such features may range from informing the agricultural practitioners, e.g., by visualizing relevant process information, to providing digital decision support, e.g., via recommendations to optimize the activities of process participants, or even implementing an automated planning and control system.

Following this nomenclature, all systems that provide digital assistance by utilizing the process information provided by an OPA system could be categorized as Online Process Support systems. However, we recommend calling them by their digital assistance system category, e.g., FMS, and denote the specific OPS features relevant within these categories, e.g., measuring the transport performance of a harvesting fleet for balancing material flow.

In agricultural logistics, an OPA system aims to identify current states of process participants and analyze the current progress of the overall processes through data analytics over the entities involved in a logistic process. As the discussion in Section 2.2.7 has shown, deriving qualitative and quantitative information about harvesting and agricultural logistic processes is essential to better inform the involved practitioners about the ongoing process and thus improve their related operational decision making. Once an OPA system can derive such information, several OPS features can be realized across all categories of agricultural digital assistance systems, including FMIS, FMS and onboard support systems.

2.3.2 Analytics Functions & Requirements

The primary question of *what* an OPA system has to do can be answered by defining the kind of data it has to provide through its data analytics functions. Some related requirements can be formulated to determine *how* these functions must or should be implemented.

Next, we differentiate between the two function types that are essential to an OPA system.

Classification Functions for Qualitative Process Information First, an OPA system shall derive qualitative process information about what the agricultural machines in an agricultural logistics process are doing. It requires suitable classification functions that ground high-level information, aggregated data, and formal knowledge by extracting it from raw data. These functions analyze incoming raw data from the process participants to find previously covert process information and make it explicit to provide high-level information about the ongoing process, as described in Section 2.5.2.

One requirement for such classification functions is that the derived data is *event-based*. This means that significant changes in a process participant's state are reflected in distinct events describing the respective transitions. Relevant transitions can thus be detected across different domains. Mainly, temporal and spatial cues can be used to trigger process-related events. For example, when a machine changes its work activity or where it changes its work location, respectively. Implementing event-based process analytics is beneficial since certain events can track significant change in the process and provide apt triggers for further data processing and the implementation of OPS features.

Another requirement for extracting qualitative process information is that the derived data should adhere to a semantic model of the process under scrutiny, cf. Section 2.4.1. Different process states are described by a distinct set of circumstances, states, and properties. This allows communicating about the current situation of the process clearly. Using a semantic model does not necessarily imply the derivation of process information in terms of formal semantic knowledge. However, opting for this is advantageous because it provides machine-readable data for further automatic processing and rule-based inference.

A qualitative analysis of an agricultural logistic processes adhering to the above principles provides continuous documentation of what, when, and where a machine was doing its work during the agricultural logistic process. The output of such a qualitative analysis can be used to visualize the ongoing process in the form of a timeline diagram, as shown in Figure 2.13.

This qualitative process information derived through an OPA system's classification functions provides the basis for all its additional analytics functions.

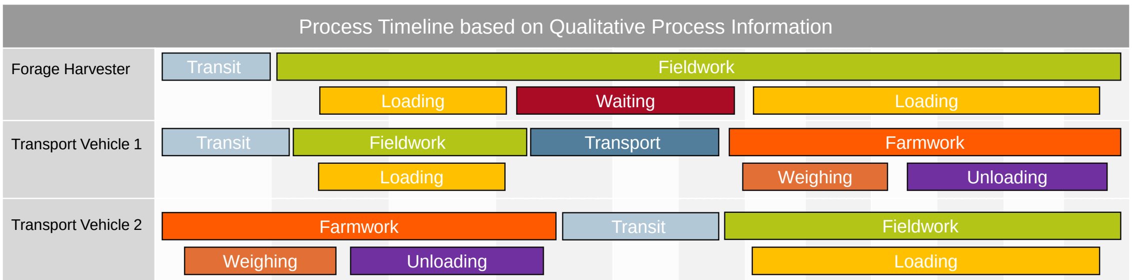


Figure 2.13: An OPA system derives qualitative process information about the states of individual process participants through classification functions. In this example, the derived information is shown as a timeline diagram that denotes the beginning and end of the different activities.

Measurement Functions for Quantitative Information The next major category of analytics functions addresses the measurement of relevant Key Performance Indicator and the derivation of quantitative information about the ongoing process. In general, the above principles of grounding high-level process information in raw data and making overt information explicit also apply to measurement function. However, they utilize various telemetry signals from agricultural machines more directly than classification functions, cf. Section 2.5.2.

Their purpose is to aggregate key performance indicators that inform the agricultural practitioners about the overall process than the individual telemetry signals. Measurement functions may summarize or aggregate specific value streams, trigger events based on value thresholds, or detect changing trends in one or multiple data streams. As a result, they can and should explicitly exploit the segmentation provided by event-based classifications, as they allow effective filtering across the temporal and spatial domain, in addition to the domains of knowledge defined in the underlying semantic models.

For agricultural logistic processes, several quantitative measurements of interest can be used to inspect the performance of individual process participants and the overall process. For example, measuring the current load of the transport vehicles is an essential measurement upon which many other transport-related metrics and evaluations can be derived. The bar chart on the left shows a comparison of estimated loads versus weighed loads, as provided by our prototype OPA system and a *calibrated* weighbridge, respectively. Taking multiple load estimations over time allows measuring the transport capacity that is utilized during the process on an average transport cycle.

Together with an analysis of how much time transport vehicles take to complete a transport cycle and how much time is spent in the different sub-processes, the transport performance of individual transport vehicles can be measured in t/h. The diagram in the center provides such a temporal analysis. It shows how many minutes a transport vehicle spent on average on the different sub-processes over the span of multiple transport cycles.

Of course, measuring these KPIs should not be calculated for individual process participants alone, but also as aggregated over entire harvesting fleets or the overall campaign. By doing so, the total current transport rates of the logistics process can be measured. If similar measurements are made for monitoring the production rates of the harvesters and the storage rate of the compactor vehicles, an assessment of the balance in the logistical network is constructed, and a high-level analysis of the logistic process becomes feasible.

The diagram on the right-hand side shows a comparison of the total harvesting, transport, and storage performance. Such a comparison is helpful to assess if the logistics process is in balance or not shows indications of how to resolve performances mismatches. In the shown example, the transport rates are too low to properly service the harvester, which in turn results in not enough material at the silo facility, such that the theoretic storage performance of the compactor vehicles is underutilized.

In summary, an OPA system enables measuring several of the essential influence factors of a forage maize harvesting process as discussed by Steckel [201].

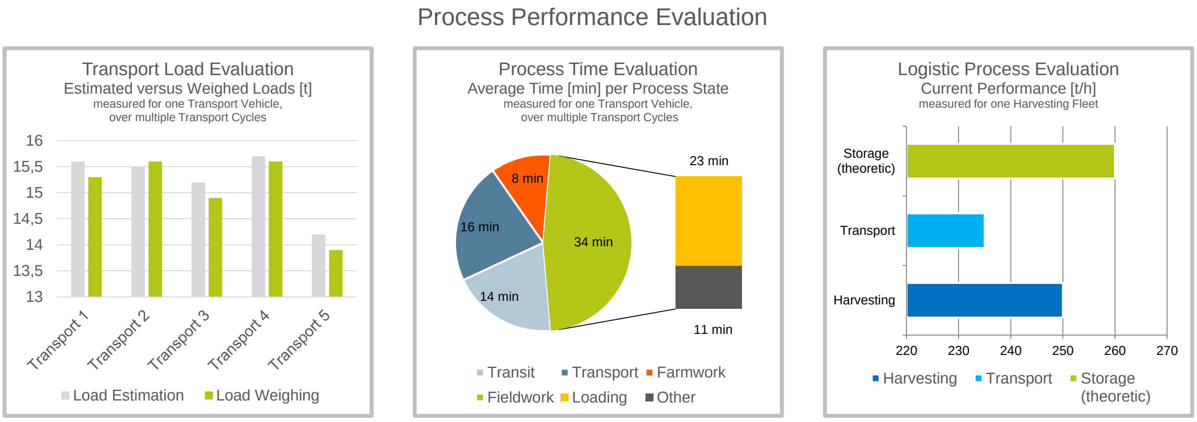


Figure 2.14: By measuring process-related KPIs, an OPA system enables process evaluation on a quantitative basis. The figure exemplifies some relevant metrics for the forage maize harvesting process, as discussed in the text. All data was taken from experiments conducted in the SOFiA project.

Process Monitoring, Evaluation and Prediction Based on these two function types and their specific implementations, the various desired capabilities of an OPA system can be developed. See Figure 2.15 for a depiction of the proposed data processing flow.

In their first application, qualitative classification and quantitative measurement functions are always applied for *online process monitoring*, i.e., to derive information and knowledge about the current state of the ongoing process. The derived data is then the input for all subsequent analytics functions that go beyond grounding current state information and provide additional aggregation and synthesis tailored towards decision support and other high-level assistance.

Among the advanced OPA functions, the category of *online process evaluation* functions concerns assessing the performance of the ongoing process. Such evaluation functions take quantitative measurements and derive a qualitative assessment of the progress of the entire process or sub-processes. To do this, they require reference values for comparison. For example, one could measure the current transport performance of a harvesting fleet against a theoretical target performance which may be determined by referencing past harvesting campaigns, drawing from best practices recommendations, or utilizing a harvest planning system. In either case, it is the OPA systems task to conduct the target-actual comparison, regardless of the target's origin.

Similarly, an OPA system can be extended to provide *online process prediction* functions, which use quantitative measurements to project metrics of interest into the future or even predict future states and events by combining qualitative and quantitative information into some form of a generative model. Here an important distinction has to be made in comparison with planning systems. An OPA system always interpolates from current states and trends forward, whereas the planning system constructs optimized targets. Hence both systems are intended to be combined such that the OPA system provides the grounding and monitoring functions, and the planning system the dedicated planning and optimization capabilities.

Support Feature-specific Analysis The above function categories are all meant to be generally applicable to all particular functions implemented for analyzing ongoing processes. While some core functions, e.g., aggregating process times, can be shared among different target processes, the particular implementation of OPA functions is always subject to the particular OPS features to be realized. Hence, additional feature-specific analytics functions or interfaces with related data processing systems may be implemented if they aid in deriving valuable process information during process execution.

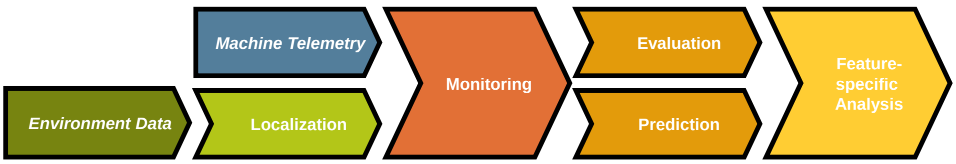


Figure 2.15: Our approach towards realizing online process analytics is based on analyzing machine telemetry in combination with environmental data. We provide input data for process monitoring functions through continuous localization, which feed into evaluation and prediction function. The combined output of all analysis functions can then be used to implement additional feature-specific analysis to provide the input for online process support features via the respective online process assistance system.

Regarding *how* an OPA system shall provide its data analytics functions in the context of digital assistance for agriculture, two major non-functional requirements can be demanded.

Near-Real-Time Capability To effectively support the execution of ongoing processes, the timeliness of data analytics is essential. For this, an OPA system shall provide its analytical output in reasonable time compared to the timing of process execution, such that the users of the respective assistance systems benefit from the OPS features and adapt their decision making and behavior based on the provided information. Therefore, data processing in near-real-time and low latencies are a key requirement for the OPA system's data analytics functions. The particular demands, however, may vary across analyzed processes and targeted OPS features. So do the limits of possible performance, depending on the complexity of the underlying analytics. Especially aggregations over time may require specific consideration.

Within this thesis, we will refrain from formulating strict latency requirements but work on the premise that an OPA system should process any input in (milli-)seconds rather than minutes.

Extendability Since many work processes can benefit from online process support, an OPA system should be designed to be applicable for many target processes. An OPA system should provide a basic set of core functions that are extensible to a wide range of processes and adapted towards the demands of a particular process and specific OPS features. Concerning agricultural logistics, any OPA solution should be equally applicable for processes with inbound or outbound logistics and cover all kinds of transport scenarios from in-field logistics to inter-regional logistics, cf. Section 2.1.2.

Technically, OPA systems should also be scalable to support many processes running simultaneously. Hence, they have to support the simultaneous connection of several machines and users. Since this requirement is mainly a matter of implementation and deployment, we will omit this subject in this thesis and leave addressing this subject to the industry specializing in digital systems for agriculture.

2.3.3 Systems Context

An OPA system can be understood to be a sub-system integrated into the system-of-systems context of digital assistance solutions for agriculture, as illustrated in Figure 2.16. Due to its declared focus on analyzing ongoing processes, it immediately addresses process *execution* and *evaluation* and thus contributes most significantly to operational decision making. Furthermore, it provides valuable input for *planning* and *control*, as well as *documentation* and thus supports all phases of process management and related decision making, cf. Section 2.1.

Regarding its implementation, it can be thought of as a purely *digital* system that needs no immediate physical extension in the agricultural process. It may be deployed server-based system or onboard of the machines. However, it is always closely tied to the *physical* machines in the real world since it requires their telemetry data to be sent in near-real-time to provide analytics functions in the same manner.

To analyze an ongoing process, an OPA system needs to be initialized with process-related information before the process execution starts. For this, it draws from *agricultural databases* and other data sources to bootstrap the analytics functions. For example, it loads information about the involved machines and their environment, the semantic process models, the targets and goals of the particular harvesting campaign to be analyzed, and other user-defined inputs. Besides the core data records, it may also require related background knowledge to provide the OPA functions with the required context information. Many of these constraints and reference data sets will come from FMS and FMIS systems. For example, the data records regarding a farmer's machines and fields can be loaded from an FMIS , while the configuration and tasks of the harvesting campaign, including production targets and work plans, might be provided by the campaign manager of an agricultural contractor via their respective FMS.

Once the system is initialized and the process started, the OPA is continuously fed with telemetry data coming from a *fleet of agricultural machines* and begins with analyzing the process. The results are then played back to the agricultural machines or other mobile devices, where this data is used to realize the OPS features that assist machine operators and process managers.

Features designed for the machine operators may provide an overview during the ongoing agricultural operations and assist in executing the work of individual sub-tasks. Therefore, they may be implemented as part of an onboard support system, a terminal application, or a mobile application on a smart device.

By incorporating the derived process knowledge into digital fleet and task management solutions, dedicated OPS features can also assist the process managers. The derived process knowledge can provide insight into the harvest's current state and continuous measurements of key performance indicators, thus increasing human managers' reaction time and decision quality during manual process control and planning. Again these OPS features may be implemented as a mobile or desktop solution depending on the manager's demands.

Additionally, the access to high-level process data in near-real-time opens up the possibility to implement a data-driven control of agricultural logistic processes by setting up a tight control cycle between an OPA and a FMS system. By directly interfacing an OPA with a digital harvest planning system, reactive planning loops or predictive control measures can be implemented, improving process optimization or even enabling automated management.

The analytics results of an OPA system can also be used within an FMIS system. For example, as input for functions such as process documentation or accounting, or provide input for post-harvest analysis and pre-planning of next year's cultivation cycle.

Following the discussion in Section 2.2.7, one could interpret an OPA system and its functions as sub-components of a FMIS or FMS systems, but how an OPA contributes insights to operational decision making is much more immediate than in the case of FMIS and FMS systems, which are dominantly concerned with planning and controlling agricultural business or work processes in terms of strategic and tactical decision making. In the scope of this thesis, we consider these systems different enough to be separated from the concept of an OPA system. However, the OPS features for process managers should be understood as a set of *novel* functions for FMS or FMIS that are explicitly tailored towards execution-related assistance and can only be realized by the incorporation of an OPA system into the system context.

Several other sub-systems have also been accounted for to fully embed an OPA system in the system-of-systems context of digital assistance solutions for agricultural processes.

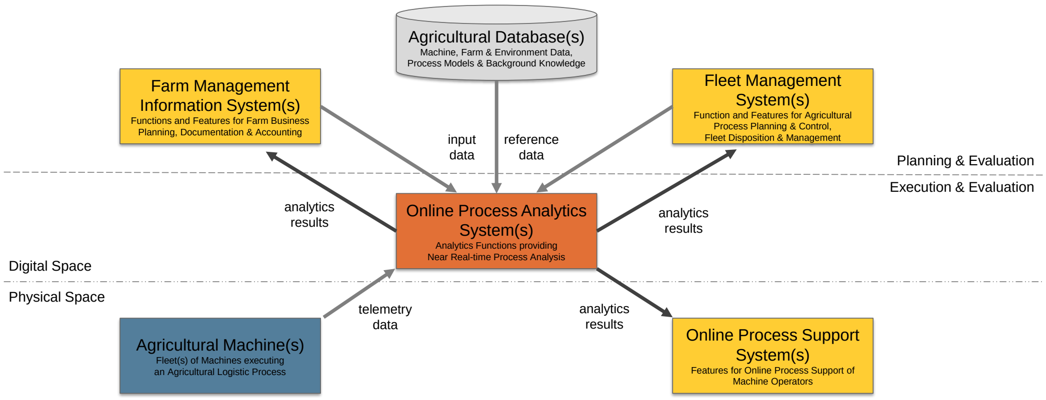


Figure 2.16: An OPA complements the already established digital assistance systems for farm and fleet management. It receives telemetry data from the machines executing the agricultural process and analyzes it to aggregate high-level process knowledge. The results can then be used for process support during execution and planning, and evaluation purposes.

The sensor systems onboard of the agricultural have to provide the data input for the OPA , cf. Section 2.2.1. It must be ensured that there is data available that allows deriving process insight from raw telemetry. Significantly, the availability of GNSS systems must be given to realize location-based OPA functions which rely on geo-referenced data.

The communication infrastructure that enables agricultural machines and assistance systems to send data to the OPA system and receive process information in return is essential, too. To support near-real-time process analytics, latency-free network solutions are ideal, but other forms of robust communication, such as delay tolerant networks, might also be applicable, cf. [130].

Considering the OPS features to be realized for the agricultural practitioners, it is further necessary to design the details of the user interfaces transporting the analytics data and the entire user experience of being digitally assisted during process execution. Similarly, the actor systems onboard of the agricultural machines need to be understood in detail when designing OPS features and related underlying OPA functions that aim to control agricultural machinery directly or even entirely automate specific (sub-)processes. Especially, when the analytics results of an OPA are used in implementing precision farming applications, cf. Section 2.2.2.

These sub-systems and their connection to OPA systems will not be discussed further within this thesis. The respective details are left to the R&D departments of companies working on commercial agricultural digital assistance systems, as solving them is mostly a matter of technical implementation and deployment.

2.3.4 Target Functions and Selected Features

We will focus the remainder of this thesis on developing process monitoring capabilities as the enabler for all advanced online analytics functions. To demonstrate the capabilities of an OPA system, we concentrate on two exemplary OPA functions and one exemplary OPS feature to be developed in this thesis. Each of the selected functions provides an example for qualitative and quantitative analytics functions, respectively.

OPA Function 1 - Event-based Tracking of the Transport Cycle To demonstrate the qualitative and event-based classification of process states in an agricultural logistic process, we aim to realize the classification and tracking of all relevant process states in the transport cycle of forage maize harvesting, cf. Section 2.1.3.1. For the transport vehicles, these process states of interest consist of the transit to and the loading in the field and the transport back to and unloading at the silo. Additionally, we want to capture the most relevant processes of the self-propelled forage harvester related to the cooperative work with the supporting transport vehicles. The concerned states cover the harvesting of the field and loading onto the transporters. See Section 2.4.1 and 2.5.2 for details on our implementation.

A more detailed analysis of the SFH's harvesting activities, e.g., differentiating between the free-form headland and track-based in-field work, is feasible based on our concepts but subject to future work. The same holds for implementing dedicated process tracking for compactor vehicles.

OPA Function 2 - Aggregation of Process Times Based on qualitative tracking, each sub-process can be timed by measuring the duration from its start event to its end event. Similarly, each complete transport cycle run can be timed once a complete sequence of subprocesses is tracked. Therefore, to demonstrate a quantitative measurement function, we chose to address the aggregation of process times since it is immediately realizable on the scope of the targeted event-based process state tracking function. See Section 2.5.2 for details on this function.

We opted for this essential measurement function since it is mandatory to develop additional evaluation and prognosis features. For example, it could be used to evaluate the stability of the transport process by analyzing the mean durations of multiple transport runs compared against the theoretic cycle times provided by a process or route planning system. Likewise, the aggregated process times could be input to predict an estimate of the expected times for future transport runs. Again, we leave detailing these functions and features for future work.

OPS Feature - Automated Documentation & Accounting Tracking the process states of all harvesters, transport vehicles, and compaction vehicles, plus measuring the time spent in their various sub-processes, provides complete documentation of the work conducted in a harvesting campaign. Providing documentation for accounting purposes is mandatory in any harvesting campaign conducted as a B2B process, cf. Section 2.1.3.2.

Therefore, within this thesis, we aim to realize an automatic documentation feature with the above analytics functions. We chose this OPS feature since it is already an established and understood assistance function in the scope for FMIS systems that can immediately benefit from online process analytics. Its details are discussed in Section 2.6.1.

Contributions to Online Process Monitoring

The remainder of this chapter will review the individual contributions of the included articles.

Section 2.4 discusses the concepts of realizing online process monitoring capabilities through spatio-semantic reasoning. Section 2.4.1 will explain why spatial analysis is essential for implementing monitoring functions for agricultural logistic processes, and Section 2.4.2 will argue for the benefit of applying semantic technology into our OPA functions. It also introduces the semantic model used to describe the process information derived by our OPA approach.

Section 2.5 summarizes our implementation of the targeted OPA functions. Sections 2.5.1 dives into the subject of spatio-semantic analysis and summarizes our work on grounding semantic maps in spatial databases. It does so without focusing on the agricultural subjects of this thesis but discusses the subject from a robotics standpoint. Section 2.5.2 builds upon our work regarding spatio-semantic analysis and applies it to the agricultural domain. It demonstrates how we implemented online process monitoring capabilities to realize the two selected target OPA functions using our semantic mapping framework.

Section 2.6 continues to showcase potential applications based on our work. Sections 2.6.1 discusses how tracking digital service contracts and service level agreements and automated documentation and accounting for harvesting processes can be realized using the developed online process monitoring functions and how we solved our selected target OPS feature. Sections 2.6.2 closes this chapter with some additional conceptual considerations regarding the architectural deployment of online process analytics systems and their role in a decentralized control architecture for agricultural logistic processes, as an outlook on future work based on this thesis.

2.4 Concepts for Online Process Monitoring

The basic assumption of this thesis is that high-level information about an ongoing agricultural logistics process can be generated by analyzing telemetry data, cf. research question RQ 1. We posed RQ 1.1, to specifically ask for a suitable data analytics method, and RQ 1.2, to clarify which additional data and background information are required for it. For answering these questions, the task of realizing online process monitoring can be broken down into three parts:

First, we have to develop suitable grounding mechanisms that allow deriving process insight from a suitable set of agricultural data. Secondly, we have to select this suitable set of available agricultural data used as input data into an online process analytics system. In Section 2.4.1, we provide an answer to these questions by presenting our concepts for realizing online process monitoring capabilities based on the spatial data analysis and the required inputs.

Subsequently, a third step puts the derived information into a format suitable for post-processing and further reasoning within a digital assistance system. In RQ 2, we asked how to represent process information in terms of formal knowledge. Section 2.4.2 presents our semantic models for the forage maize harvesting process that allows representing the process information derived by our process monitoring solution. The results contribute to answer RQ 2.1.

2.4.1 Process Monitoring through Spatial Analysis

At the core of our analytics approach is the insight that agricultural logistic processes are executed by a set of dynamically moving and spatially distributed agricultural machines.

Spatially Distributed Sub-Processes of Harvest Logistics In [1], we worked on the idea of delegating process-related decisions to the individual distributed process participants, cf. Section 2.6.2. Moving from centralized decision making to a distributing setup leads to the requirement that the agricultural logistic process is segmented into independent sub-processes, which can be decided about by individual process participants or by cooperating pairs of process participants. Our developed model provides a hierarchical structure of the individual sub-tasks of forage maize harvesting, as shown in Figure 2.17. For building this model, we exploit that forage maize harvesting is a spatially distributed process and propose two semi-independent decision-making hierarchies for harvesting the field and compacting a silo at the farm, respectively, which are linked by the transport process.

Based on our model, we assume that the spatially distributed nature of the process can be exploited for monitoring, too. We observe that significant changes in the whereabouts of individual machines can be a crucial discriminator to determine what each machine is doing if a stable mapping between process location and process states can be found. Hence, we analyzed the sequence in which the sub-tasks of forage maize harvesting are executed and analyzed which spatial events provide the cues that allow distinguishing the process steps. This was done for all machine types. Most individual process states coincide with distinct spatial events at the arrival or departure at a particular location or partner machine.

Figure 2.18 shows the result of this analysis for transport vehicles. It illustrates how the individual sub-tasks of the transport cycle are immediately separable due to the distinct change in location and road travel in between. By grounding events such as *arrival at the field* or

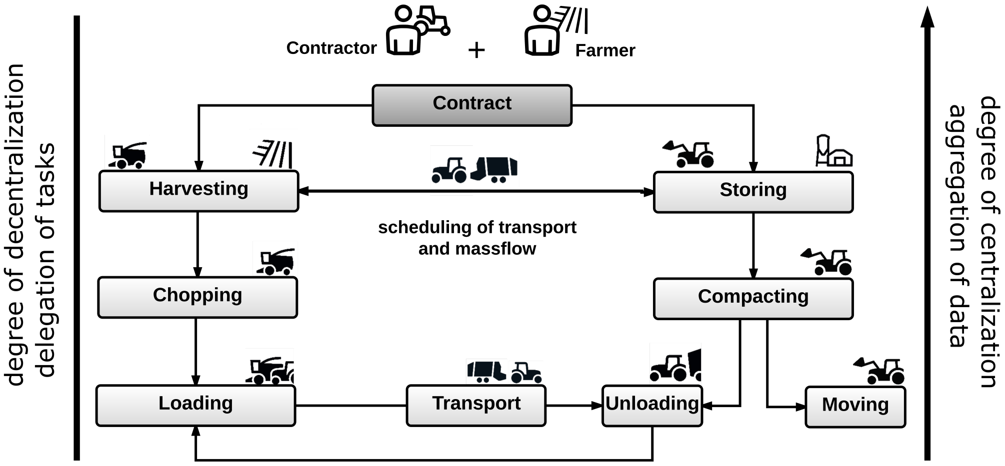


Figure 2.17: For decentralized control, the harvesting process is segmented into independent subprocesses, which the machines plan and execute as jobs.

departure from the farm and tracking the relations between these events over time, relevant insight into the process can be derived. We, therefore, base our process monitoring approach on a qualitative classification function that continuously determines the machine's spatial location, cf. Figure 2.15. The function's output provides knowledge about the topological location of all machines in terms of spatial predicates and localizes agricultural machines in their environment. It is a grounding function for machine-environment relationships.

Monitoring the close collaboration between harvester and transport vehicles during loading requires additional spatial analysis because localization alone does not provide insight into machine-machine relationships. But again, spatial analysis can provide the necessary cues to detect the beginning and end of loading operations. For example, inspecting the spatial distance between pairs of machines can identify if the transport vehicle is in close proximity to the harvester, such that harvesting is possible. To derive additional insight into the loading, e.g., determining the transferred loads, additional analysis over the harvester's telemetry is required. See Section 2.5.2 for related details.

In summary, we need a set of grounding functions to create an event-based data stream that captures the spatial relationships of the participants of a harvesting campaign with respect to their environment and each other. To classify process states based on this stream of localization events, we intended to inspect it for significant localization changes over time and map these to process-related events.

Grounding Spatial Relationships in Agricultural Environment Maps To realize this continuous spatial analysis, our method requires positional information of the agricultural machines and an model of the relevant agricultural environment to match against.

Because modern agricultural machines are commonly equipped with GNSS receivers, positional information is part of the telemetry. Since mobile connectivity is common on modern agricultural machines, we can assume that the machine telemetry can be sent as a continuous data stream

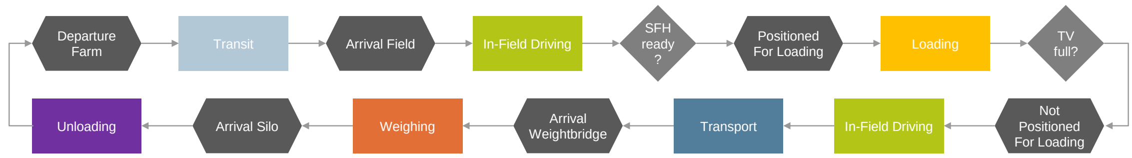


Figure 2.18: To service a harvester, a transport vehicles cycles between farm and field. For identifying the different process states, it is helpful to analyze the spatial relations between the transporter, its partner harvesters, and the environment because these provide event-based cues to separate one process step from another. The flow chart illustrates the spatial relations used in our prototype system.

from the machines to a backend system. For the sake of simplicity, we assume full connectivity between the agricultural machines and the OPA system, such that recorded telemetry data is available for analysis in near-real-time.

Next, this telemetry data stream must be analyzed in reference to spatial information about those parts of the environment relevant to tracking process states. For a forage maize harvesting process, the locations of interest are the field, in which the harvest takes place and the silo facilities to which the harvested material is brought. Additional locations of interest may include the farm in general, as well as the weightbridge.

As discussed in Section 2.2.3, mapping agricultural environments is also common practice in the state of the art. For a basic localization function, simple 2D spatial maps of the locations of interest in a geo-referenced format suffice for a direct matching against the machines GNSS positions. It can also be assumed, that during the execution of a harvesting process, the spatial properties of the agricultural environment are assumed to be static, such that the respective data can be loaded before the process starts, e.g., by interfacing an FMIS.

For our prototype, we utilized basic 2D spatial maps, as shown in Figure 2.19, and we did not consider any other environment maps and other related data.

Additional spatial data can also be incorporated into a localization function. For example, street maps and other navigational information, e.g., traffic data, could inform the machines during road travel. Likewise, detailed agricultural maps could be used for process evaluation functions within an OPA system, e.g., yield sensor readings could be related to an map of expected yields. Similarly, environment data, such as weather forecasts, are a well suited complement for qualitative localization functions with quantitative measurements and analysis regarding the machine and its environment.

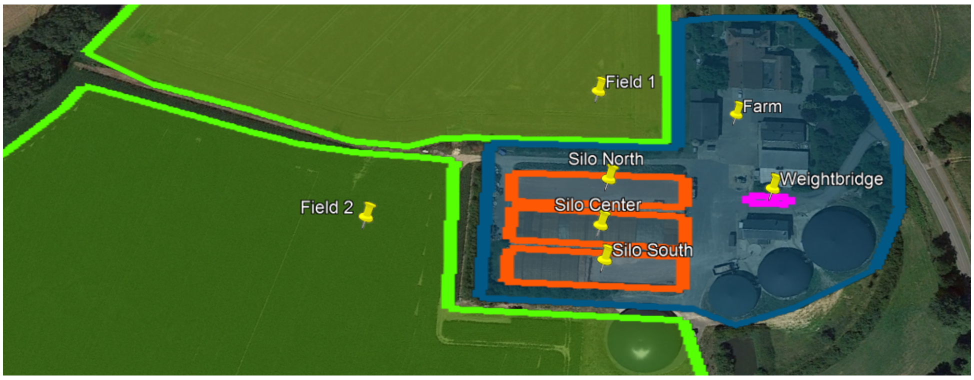


Figure 2.19: To represent a farm's facilities in SEMAP, we used the 2D polygonal boundaries and stored them in the spatial database component. These spatial models are connected to instances of the domain-specific concepts of the AgriCo ontology in the knowledge base component.

2.4.2 Semantic Models for Forage Maize Harvesting

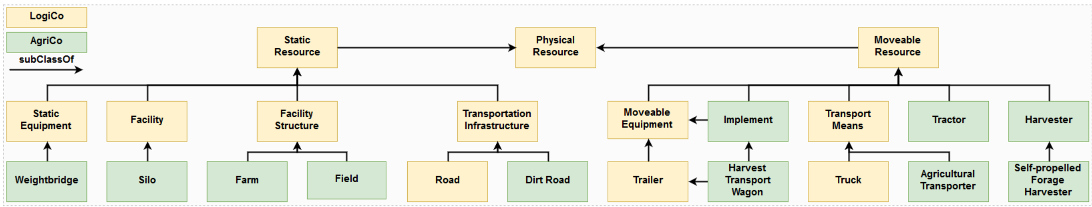
In Section 2.4, we formulated the requirement, that an OPA system should produce process information in terms of events that adhere to a semantic model of the process. This means that the types, states and properties of an agricultural logistics process are defined and described in detail. The OPA system can use this process description during process analysis. Its purpose is two-fold: First, it serves as a reference point for the classification functions and provides the structural patterns whom the process is likely to follow. Matching detected states against the model gives a sort of blueprint for validation.

Secondly, the model defines the relations that hold between entities and describes their properties. This is important, as it clarifies what certain measurements derived by the OPA system's measurement function tell us about the process.

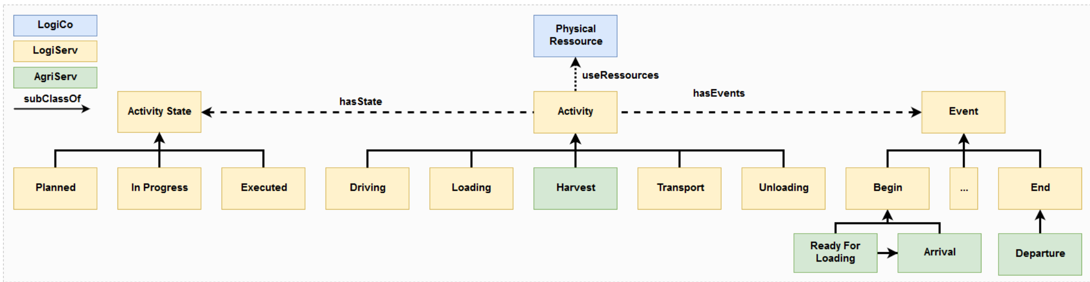
Domain-specific Ontological Model We chose to utilize formal semantics for our prototypical implementation such that the derived process knowledge is in a machine-readable representations that enables ontological and rule-based reasoning. All ontological models and derived facts about the environment were represented in the Web Ontology Language (OWL) [24].

We designed a semantic representation of the entities involved in an forage maize harvesting process. We presented the resulting AgriCo ontology as a semantic model for agricultural machinery and their environments in [79].

In that work, we reviewed ontological models from classical logistics and based our model on the Logistics Core Ontology (LogiCo) by Daniele et al. [73]. Our model extends this semantic model of environments and resources in logistics, with the additional concepts needed to represent entities specific to the agricultural domain. For example, farms and silos were added to AgriCo as facilities that serve as static resources. Likewise, the basic concepts of movable resources and transport means were extended by agricultural concepts, such as transport vehicles and harvesters, as shown in Figure 2.20 (a).



(a) Excerpts of the domain-specific semantic model used to describe agricultural logistic processes. The LogiCo ontology provides a basic model of logistic resources, to which the AgriCo ontology adds concepts of the agricultural domain.



(b) An excerpt of the AgriServ ontology that provides an event-based model of agricultural logistic processes. It extends the LogiServ ontology and builds upon the AgriCo and LogiCo ontologies.

Figure 2.20: Excerpts of the ontological models providing the vocabulary for a semantic description of process knowledge derived through the online process monitoring functions.

Modeling Event-based and Process-related Information Next, we needed to define the ontological vocabulary to capture state changes and describe the events to represent the classification results of both our localization and process monitoring functions. To explicitly incorporate concepts about the harvesting process in our semantic model, we created the AgriServ ontology that allows us to describe agricultural work and services in terms of the activities that have to be performed to achieve a specific logistical objective in the agricultural domain, as shown in Figure 2.20 (b). It was derived from previous works in logistics [73, 114] and extends our AgriCo ontology to refer to the resources involved in the various process states and events.

Within the model, activities are described as a sequence of events and hold a list of associated instances. Multiple events and states can be related to a particular activity instance. Furthermore, these relationships are differentiated by sub-relations that denote if an event is scheduled, i.e., belonging to a planned activity, or whether it is ongoing, to describe the events occurring during process execution. Following our focus on spatially derived knowledge, we also introduced dedicated events describing spatial state transitions of a movable resource, such as arrivals and departures from a facility or the domain-specific state `ReadyForLoading` derived from the loading-related classification functions discussed in Section 2.5.2.

In summary, the AgriCo and AgriServ ontologies provide an semantic model to describe an agricultural logistic process and its participants, and, thus, meets the requirements regarding an event-based semantic process model, as formulated in Section 2.3.

2.5 Implementation of Online Process Monitoring

The previous chapter summarized our concepts for realizing online process monitoring capabilities using spatial analysis in conjunction with semantic technology. This section further explores the principles of spatio-semantic reasoning and details how we implemented these concepts in a prototypical software system capable of providing the selected process monitoring functions.

Section 2.5.1 focuses on addressing research question RQ 3 and explains how semantic information about spatial relationships can be derived from semantic maps (RQ 3.1 and RQ 3.2). It also shortly summarizes our work on the SEMAP system, which is at the basis of our prototypical implementation. All content was as originally published in [3].

In Section 2.5.2, we continue with summarizing the results of [4] regarding our implementation of online process monitoring capabilities by using the SEMAP framework. The presented work is directly based on the conceptual considerations in Sections 2.3 and 2.4, and further contributes to answering RQ 1. It also relates to RQ 2 and RQ 3, as it presents examples of spatio-semantic inference in the agricultural domain, which directly answer to RQ 2.2 and RQ 3.2, respectively.

2.5.1 Spatio-Semantic Inference over Semantic Maps

This section summarizes our work regarding spatio-semantic reasoning in the context of semantic maps for mobile robots, as published in [3].

This article presents how to derive and manage qualitative spatial relations between objects from quantitative geometric environment data captured by some mapping approaches. It demonstrates effective spatio-semantic querying on semantic maps by integrating a spatial database into a semantic mapping framework. The proposed method is based on the close integration of a spatial database that provides a dedicated storage and processing module for the spatial environment data as a suitable complement to a classical knowledge-based system.

Our work is based on a review of several approaches in semantic mapping. We find that significant progress had been made in describing the semantics of environments using ontological models to capture a-priori background knowledge and facts about an environment's current state. It also shows that mature approaches in spatial mapping, scene segmentation, and object recognition allow gathering spatio-semantic data of real-world environments at large scales.

The review further shows that it is crucial to link semantic knowledge with geometric data and perform analysis across both domains, especially, when a robot's multi-modal environment representation must be updated, e.g., when new information is acquired. However, it also reveals that this essential grounding of spatial relations is commonly done during semantic map building and then statically encoded within the map. We find this lacking, as it makes semantic maps unnecessarily rigid and prevents deriving grounded spatial knowledge from the gathered environment data on demand. It is apparent that a build-in grounding mechanism on the semantic map's representational level is required to transform the gathered environment data (regardless of it being semantic knowledge or quantitative geometric measurements) into a generalized multi-purpose model.

We, therefore, address the issue that the representational frameworks underlying semantic maps are still unable to ground spatial relations between entities within our work. Our focus is to derive a semantic map representation able to analyze spatial relations in terms of qualitative predicates, as this is important in data retrieval and reasoning. To fully utilize qualitative spatial reasoning, it is necessary to derive qualitative symbolic data from quantitative metric information. Therefore, we aim to integrate dedicated tools for performing spatial analysis on quantitative metric data into a semantic map representation.

Core Concepts In [3], we propose to pair spatial databases and declarative knowledge bases to combine ontological and logical rule-based inference with spatial querying and analysis capabilities. We also demonstrate our concepts by implementing the SEMAP framework to represent, manage and query spatio-semantic environment data.

The framework aims to provide information about the objects and the environment in a specific application domain while being domain-agnostic in its technical representation. It connects conceptual knowledge about the environment and factual knowledge about present object instances with their geometric representations to hold a combined spatio-semantic model that allows spatial analysis and semantic inference. SEMAP internally separates environment data into two dedicated databases to ensure optimized performance for each modality to manage the fundamentally different structure of semantic and spatial information. It also provides a multi-modal interface to link between associated data records during insertion and retrieval.

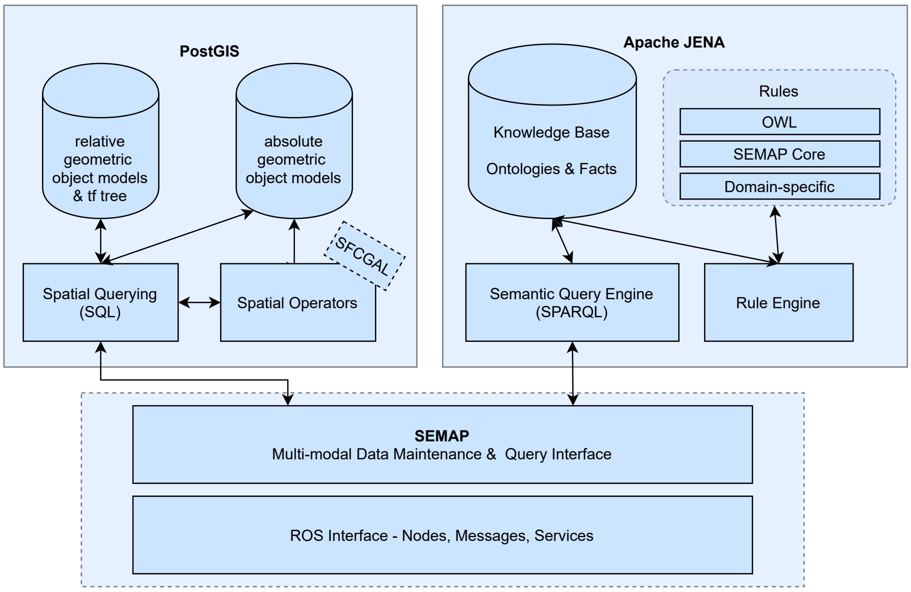


Figure 2.21: SEMAP combines a spatial database with a knowledge base, such that spatial data and semantic knowledge is stored separately in dedicated components. A query interfaces both storage solutions to provide joint access to the represented environment data and delivers it on demand.

An outline of SEMAP's internal structure is given in Figure 2.21. The semantic part is represented by a knowledge base system (KB) based on description logics with the obligatory separation into terminological and asserted knowledge. The environment's conceptual model and facts about the environment are represented in the Web Ontology Language (OWL) [24] and maintained in Apache JENA [58, 147, 210], which provides inference for ontological and rule-based reasoning as well as the capability to query the stored knowledge. The spatial part is a dedicated spatial database system (DB) that stores geometric primitives and provides operators for quantitative spatial analysis and spatial querying. It is implemented using PostGIS [168].

Geometric Representation & Spatial Querying All geometric information is stored in the spatial database and describes the individual objects' shapes and poses in the environment. For articulated objects, their kinematic chains and current joint configurations are represented as well. Additionally, the database maintains relational links that connect geometric data sets to their complementary semantic descriptions and facts in the dedicated knowledge base.

Spatial querying and analysis are done using SQL queries that incorporate spatial operators implemented within the spatial database. SEMAP provides a range of metric, topological, and directional spatial operators for both 2D and 3D analysis, cf. Section 5.3.5. These operators are used to compute qualitative spatial facts about the maintained objects, cf. Section 5.3.6.

Ontological Model & Rule-based Reasoning SEMAP incorporates a classical knowledge base component to represent formal knowledge and enable rule-based reasoning over this knowledge. It uses description logics [16] and OWL-based ontologies.Using ontologies makes SEMAP's core components remain application-independent and extensible to different application domains. The T-box maintains a set of domain-independent ontologies that provide a semantic model of the supported geometric types and encode how they can be combined to form objects and how a set of objects constitute an environment. In addition, a set of domain-dependent ontologies can be inserted to provide the necessary vocabulary to describe knowledge about a particular application. Within the A-Box, the combined ontological descriptions store facts on the object instances. SEMAP allows for inference over the stored knowledge using ontological reasoning and rule-based reasoning by combining SWRL rules [113] with SPARQL queries [231].

The semantic description of environment entities in SEMAP is constructed such that entries within the spatial database are closely linked to the respective facts about these instances. See Figure 2.22 for a sketch of the core ontology. It uses standards from the Open Geospatial Consortium as they are well-defined and widely adopted models for representing geo-spatial data. To join spatial and semantic data, SEMAP uses a set of associations to provide a consistent spatio-semantic object model, see Sections 5.3.3 and 5.3.4 for details.

Domain-specific ontologies, knowledge bases, and rule-sets can be imported into SEMAP to create a spatio-semantic environment model for a particular application. To describe domain-specific concepts spatially and reason about them as part of the environment model, the respective entities can be associated with object models stored in SEMAP to utilize both their semantic and spatial representation. In summary, SEMAP allows for a rich semantic environment description with a full geometric representation that can be queried across both modalities.

Spatio-semantic Reasoning & Domain-specific Applications The framework's strength lies in combining both query systems to support combined queries with semantic and spatial aspects like *"Is there a computer in this room?"*, *"Which mug is the closest to the robot?"* or *"Is there a keyboard in front of the monitor?"*, cf. Figure 5.1. In such queries, SEMAP uses the DB's spatial operators to ground qualitative spatial relations that were previously only implicitly

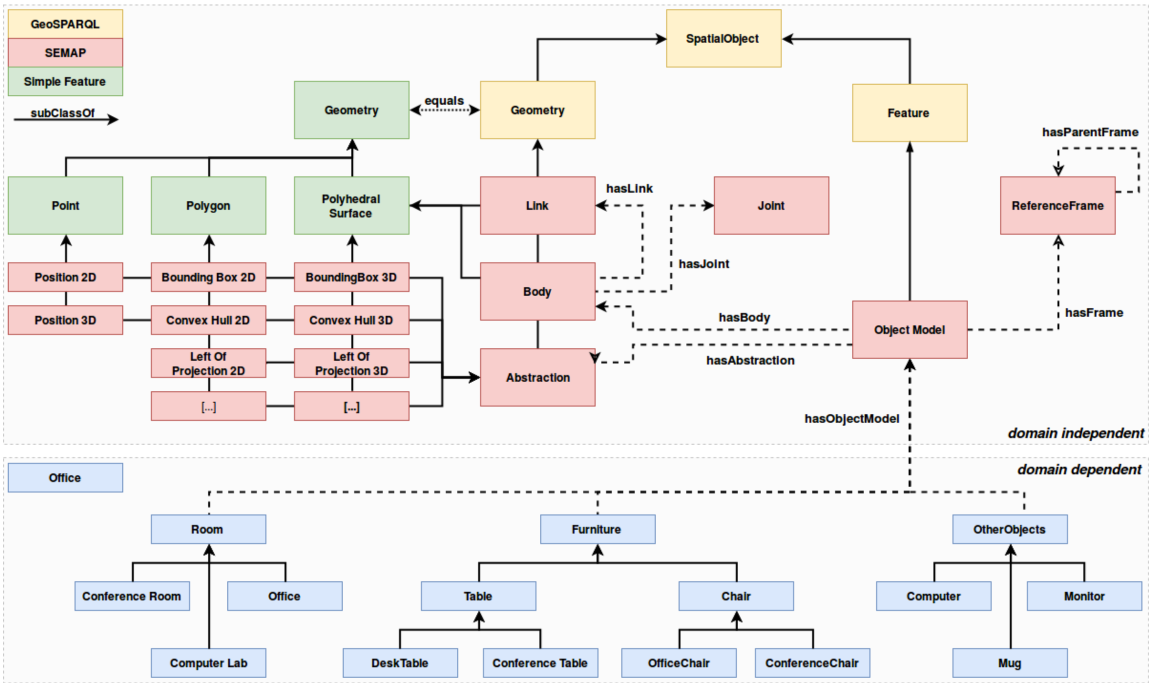


Figure 2.22: An excerpt from the ontology that is used to fuse the labels of objects stored in the spatial data base with semantic concepts in different domains.

stored in the geometric environment representation to make them explicitly available for semantic queries, cf. Figure 5.18

For applications working with the framework, it provides a combined query interface that translates into these languages and maintains the synchronization between both representational components, cf. Section 5.3.6. With each quantitative spatial query over the DB, new facts on qualitative spatial relations are generated and automatically inserted into the KB as facts for further inference. This approach enables rule-based reasoning and constructing complex spatial queries based on more straightforward deductions.

This multi-modal query interface is advantageous in real-world applications, as it allows to answer complex questions about the positions, relations, and roles of the stored objects in a natural way. Hence, SEMAP is able to topologically structure environment information, implement qualitative scene classification, or support object retrieval tasks. In the article, we argue that an environment representation based on the above concepts can easily provide the specific environment information needed for a particular application and exemplify this in the context of controlling a mobile robot within an office domain. See Section 5.4 for examples and answers to RQ 3.3 in a non-agricultural application context.

2.5.2 Process Monitoring through Spatio-Semantic Inference

The following section summarizes how we realized online process monitoring capabilities for an ongoing agricultural logistics process through spatio-semantic inference.

As discussed in Section 2.4.1, we based our approach on the insight that harvesting processes are spatially distributed and the assumption that process states can be classified by applying spatial analysis in conjunction with other analytics means. Likewise, we argued for the benefits of using knowledge-based systems to capture agricultural process information and presented a semantic model for forage maize harvesting, cf. Section 2.4.2. In Section 2.5.1, we discussed the core concepts of spatio-semantic reasoning and introduced the SEMAP framework as a tool for representing environment information both spatially and semantically and for querying and reasoning across both modalities.

This section explains how we applied SEMAP's reasoning capabilities to implement the selected target function, proposed in Section 2.3.4, as a proof of concept for our considerations on online process analytics for agricultural processes.

Applying SEMAP in the Agricultural Domain As discussed in Section 2.4.2, the ability to localize agricultural machines in their environment and analyzing spatial relationships between cooperative machines is essential to our process monitoring approach. We chose to realize these grounding functions using our semantic mapping framework SEMAP, as its features aptly provide the basis for implementing these grounding functions.

In our original work on SEMAP, we used the system to represent a mobile robot's environment to let it reason about it. We continuously update the robot's position within the environment and dynamically queried for information about its surroundings. Hence, we utilized SEMAP for the benefit of a single actor in the represented environment. However, the framework allows for more actors, too, so we decided to introduce multiple dynamic actors, i.e., the fleet of machines conducting the forage maize harvesting process.

The framework also allowed us to incorporate our semantic process models and represent and store the process knowledge derived through our monitoring system.

Instantiating the Agricultural Model Based on our semantic model, we continued to instantiate a spatio-semantic environment representation as a dataset maintained in SEMAP. First, we imported the AgriCo ontology into SEMAP's KB component, then we added facts about particular entities to populate our example scenario with transport vehicles and harvesters in an environment of fields and a farm.

Next, we added the spatial representation. To set up static resources in our environment model, we used a set of polygonal boundaries to represent farms and fields and other facilities, cf. Figure 2.19. To represent the movable resources, we created three-dimensional and articulated object models of a tractor-trailer combination and a forage harvester as displayed in Figure 2.24 (b). Figure 2.23 illustrates how the domain-specific ontology is tied to the core concepts of SEMAP and how facts about instances of domain-specific concepts can thus be linked to their spatial representations. Section 6.4.2 explains this linkage in detail.

Since we wanted to utilize spatio-semantic reasoning for the ongoing agricultural logistic processes, it was further necessary to introduce dynamic motion into our example scenario. To move and articulate the modeled agricultural machines, we replayed recorded telemetry data, including GNSS signals and joint states. This replay was realized through the Robot Operating System (ROS), which we connected to SEMAP, such that the environment model was updated accordingly. Our test data was recorded on real machines during a harvesting campaign, which we attended with the SOFiA project, cf. Sections 7.1 and 7.3.2.

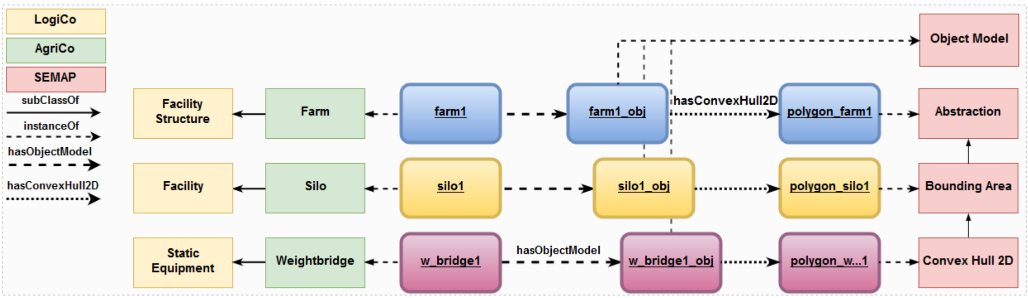


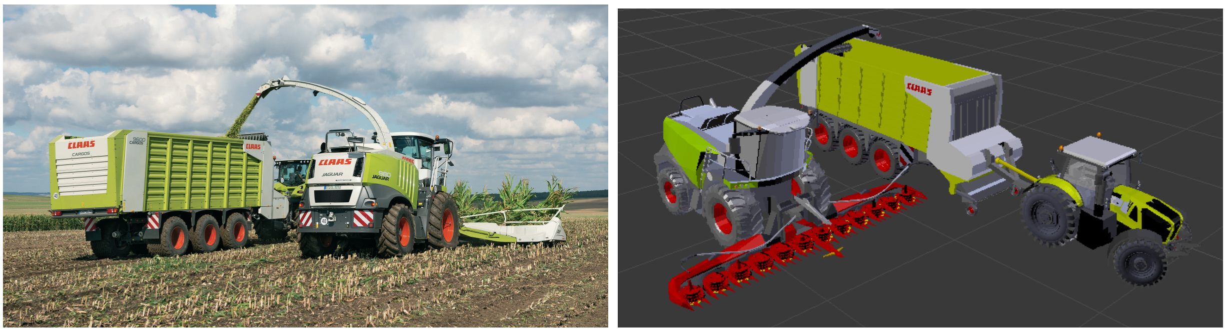
Figure 2.23: To link spatial data to instances of the domain-specific concepts of AgriCo, we used SEMAP's `ObjectModel` concept and its relations.

Inferring Spatial Machine-Environment & Machine-Machine Relationships Through this setup, all the spatio-semantic queries provided by the framework were made available for the entities represented in SEMAP. We proceeded to apply the available inferences provided by SEMAP to realize the required localization and process monitoring functions.

We used the framework's query system, for example, to infer the topological location for all pairs of machinery and environment entities, based on the spatial containment of the machine's GNSS position in the polygonal boundaries, too. Within the article, we argued that while this seems like a simple transition, the used rule effectively infers from a *spatial* predicate to a *topological* relation and that this *semantic* assertion is grounded in the quantitative *geometric* data within SEMAP's DB. It also provides the necessary grounding of the spatial machine-to-environment relationships for our localization function. Since the underlying rule is generic for all instances of movable resources at any facility instance, it is applicable in a wide range of logistic applications, not only agriculture.

Similarly, we used the same type of reasoning to analyze spatial relations between pairs of machines. For example, we constructed a query to detect that a transport vehicle is correctly positioned for a loading procedure due to its directional relations regarding a harvester. Here we leveraged SEMAP's ability to represent articulated objects in 3D and the 3D spatial operators of SEMAP's spatial query system. Rules, like the one shown in Figure 2.24, exemplify how to construct complex domain-specific relations by combining several fundamental spatial relations with additional domain-dependent knowledge.

Within the included article, we argued that this approach could be used to construct various different *grounding functions* to generate particular facts about the relevant states occurring during a harvesting process. As an example, the above grounding procedure could incorporate the joint values for the loading boom to precisely determine *where* the harvester's material flow meets the transport vehicle's trailer as proposed by Happich [112].



(a) Loading in reality. **(b)** Loading in RViz.

```
?sfh rdf:type agrico:Harvester
?sfh semap:hasObjectModel ?sfh_obj
?sfh_obj semap:hasLeftOfProjection2D ?sfh_proj_l2D
?tv rdf:type agrico:TransportVehicle
?tv semap:hasObjectModel ?tv_obj
?tv semap:hasConvexHull2D ?tv_abstr_ch2D
?tv_abstr_ch2D semap:isIn2D ?sfh_proj_l2D
==>
?tv agrico:positionedForLoading ?sfh
```

(c) The rule for grounding the `positionedForLoading` relation in SEMAP.

Figure 2.24: We used telemetry data from an actual loading procedure (a), to move and articulate the machines in ROS and visualize them in RViz (b). We also synchronized the telemetry with our SEMAP model and used the rule (c) to identify the correct spatial positioning of two machines for loading harvested goods from a forage harvester onto a transport vehicle.

Classifying Process States By continuously applying the implemented grounding functions over the changing environment representation, it became possible to detect state changes and derive an event-based tracking of the unfolding process in line with our proposed concepts.

Since SEMAP's query system analyses the relationships within its environment model as facts that hold at a particular point in time, we continued our experiments by querying the system for relevant relations with every incoming telemetry datum. By inspecting the changing spatial relations in our application example, we generated a continuous trace log of the machines' whereabouts and their relations towards each other in terms of qualitative spatial relations.

Since these spatial transitions give a strong indication of the underlying agricultural process, we created a stream of spatial events using the concepts and relations provided in our AgriServ model, cf. Section 2.4.2. When, for example, the fact `tractor1 isAt farm1` did hold at timestamp t_n, but does no longer hold at t_{n+1}, an `Departure` event is created and asserted to the KB. By immediately applying the ontological and rule-based reasoning over these grounded spatial facts, we derived the high-level process states and events describing the harvesting process. Figure 2.25 shows the mapping from spatial onto process events. In the example, the spatial arrival of a harvester at the field triggers the beginning of the process state `Fieldwork` and the beginning and end of the spatial relation `positionedForLoading` is used to ground the start and end of a `Loading` procedure between harvester and tractor.

# Time	# Reference	# Spatial Relation	# Event	# Target
13:16:51	harvester1	onField	Arrival	field2
13:17:02	tractor2	onField	Arrival	field2
13:17:36	tractor2	inDistance	Arrival	harvester1
13:17:45	tractor2	positionedForLoading	Arrival	harvester1
13:20:21	tractor2	positionedForLoading	Departure	harvester1
13:20:29	tractor2	inDistance	Departure	harvester1

(a) Spatial Events

# Time	# Reference	# Process Type	# Event	# Target
13:16:51	harvester1	Fieldwork	Begin	field2
13:17:45	tractor2	Loading	Begin	harvester1
13:20:21	tractor2	Loading	End	harvester1

(b) Process Events

Figure 2.25: A continuous trace log of spatial relations between machines and environment created through analyzing telemetry data with SEMAP was mapped onto the process states and events denoted in the AgriServ ontology.

To cover all process states defined in our semantic model, we constructed several spatio-semantic grounding functions and rules for mapping from spatial predicates to semantic facts about process events. For proper event generation, we added an additional processing node to our prototypical system, which accounts for the state history of each dynamic entity and generates the appropriate events if a state transition occurs by comparing states with different timestamps. This extension was necessary since the SEMAP system itself provided no fundamentals to represent time, track changes over time or conduct temporal reasoning.

Measuring Process Times & Tracking Process Sequences Next, we extended our process monitoring component to construct not only process events holding for a particular timestamp but to produce facts that hold true for a specific time interval too. These process state intervals can be constructed by determining pairs of associated events that determine the beginning and end of a fixed event type occurring in the event sequence of a particular agricultural machine. When the respective interval is closed, this is detected by the processing node, and an additional process event is emitted.

This construction of interval-based process states allowed us to reason about the process durations and other key performance indicators. For example, access to the harvester's telemetry enables us to link a current measurement of the machine's total yield counter to the beginning and end events of a particular loading sequence. When constructing the respective process interval, it is then possible to estimate the total mass loaded onto the transport vehicle.

We continued to apply the same style of reasoning towards reasoning about more complex activity patterns. As discussed in Section 6.4.6, we constructed inference rules that detect more abstract, not immediately spatially related process states of transport vehicles by applying template matching over sequences of process intervals. For example, we provided a rule to identify a full transport cycle between field and farm by analyzing temporal succession of process intervals for loading, transport, unloading and driving.

In summary, we combined the SEMAP system with the AgriCo and AgriServ ontologies. We utilized this setup to demonstrate the conceptual feasibility of spatio-semantic reasoning for monitoring ongoing agricultural logistic processes. Through this approach, we solved the two monitoring functions we defined as targets in Section 2.3.4. Based on this prototype solution for process monitoring, we continued to apply the derived process information into an agricultural application.

2.6 Applications of Online Process Monitoring

In the previous sections, we have presented our concepts for spatio-semantic reasoning and how it can be applied in the agricultural domain to implement online process monitoring capabilities in the context of an online process analytics system, cf. Sections 2.4 and 2.5.

Within this section, we address Research Question RQ 3.3 that asked: *How can spatio-semantic reasoning be applied in a real-world domain?*

To answer this question, we draw from the included articles reproduced in Chapters 4 and 7. Firstly, we review our solution to implement process documentation and automated service tracking and accounting, based on our online process monitoring capabilities, as published in [5]. Secondly, we discuss how the subject of online process analytics, in general, and the implemented online process monitoring capabilities, in particular, can contribute to decentralized approaches of planning and controlling agricultural logistic processes, as published in [1] and [2].

2.6.1 Automated Service Tracking & Accounting

Process documentation and accounting are standard features for Farm Management Information System, where they are usually solved in post-processing. Given our online process monitoring functions, we can derive the relevant information for documenting a harvesting process during execution. Hence, it is possible to implement business-related service tracking as a near-real-time feature and quickly provide automated accounting.

We, therefore, chose process documentation as the target OPS feature in this thesis to demonstrate the applicability of online process monitoring data in an extension to this feature.

In [5], we worked on digitizing business processes related to logistic processes in agriculture. Our work was motivated by the SOFiA project's goals of contributing to *smart finance* solutions for logistics, cf. Section 7.1. Following the recent trends of increased digitization in many industrial sectors, the project focused on digitizing service-related tasks in the logistics sector.

Within the included publication, we studied the business interactions between farmers, contractors, operators of biogas plants, and financial service providers around the agricultural process of forage maize harvesting. As discussed in Section 2.1.3.2, many related business transactions are still managed informally and thus provide much potential for improvements.

Improvements to the Harvest-related Business Processes As an example use case, we assumed a scenario in which a commercial biogas plant (BGP) cooperates with a set of farmers and an agricultural service contractor as business partners, cf. Figure 2.26.

BGPs need a large amount of fermentation substrate over the year, usually more than the own agricultural business can provide. Therefore, it is common to utilize sub-contractors. The cultivation of maize plants is outsourced to multiple farmers, who cultivate their fields and then sell the finished maize plants to the BGP business. Regarding compensation, the harvested maize is charged per tonne of dry matter. Hence each delivery to the silo must be assigned to the correct field and thus to the farmer as a contractual partner to enable correct accounting. Additionally, a weighbridge determines how much net weight of maize chaff a transport vehicle has unloaded at the silo. The performance acceptance is done manually, and complex control mechanisms (e.g., a four-eye principle) must be integrated to ensure error-free documentation.

Keeping a “weighing record” thus generates considerable effort for the operator of the BGP and conflicts with the other tasks that arise during the process runtime, such as the scheduling of the transport logistics. This procedure requires the farmer’s trust towards the operator of the BGP, as the documentation is carried out by employees of the BGP. Therefore, the service documentation is the service recipient’s responsibility and thus not entirely transparent for the farmer as the service provider.

The execution of the harvest and the production of the silage is also subcontracted. An Agricultural Contractor (AC) provides the necessary machinery and workforce. Often, the entire harvesting campaign across all fields of the supplying farmer is outsourced to the AC. During the harvest, the AC is in close contact with the BGP’s operator and the individual transport vehicles to coordinate the harvest and its logistics. Work hours of machines and operators are recorded to settle the services provided by the AC with the BGP. Depending on the contract, the areas worked, the kilometers driven and the fuel consumed is also recorded and settled. This documentation is handled by the AC’s employees who operate the agricultural machines and carry out the harvest. As before, the same problems regarding performance acceptance arise. It is in-transparent to the BGP as service recipient because the AC as service provider documents his own performance. It is also prone to errors because proper documentation is often neglected in the face of the actual agricultural work.

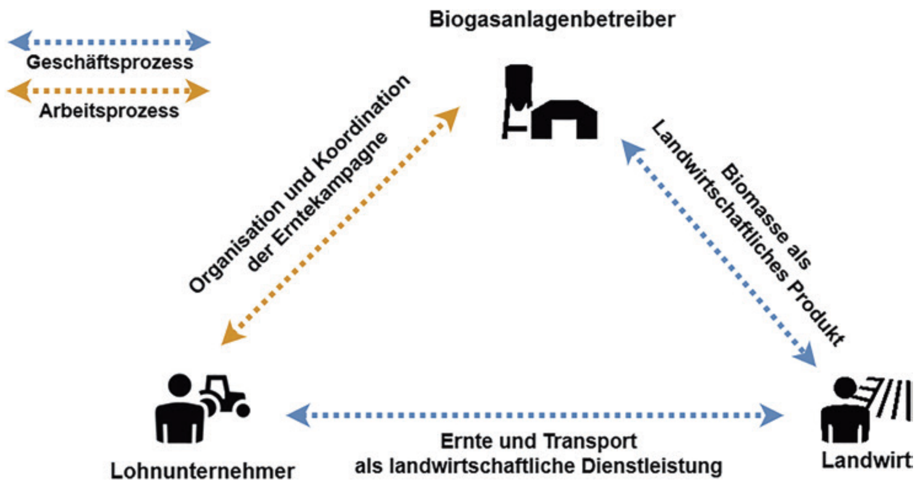


Figure 2.26: Business relationships in the agricultural use case

Following the agricultural work process, one could assume that the AC charges the BGP operator for the total cost of maize harvest and silage production just as the farmer might charge the BGP operator for biomass production. However, this is often avoided to secure entitlement to agricultural subsidies (e.g., on diesel). To make good use of the economic advantages of agricultural work subsidies, the contractual setup of business relations strongly deviates from the actual work organization. The AC therefore bills the individual farmers directly, as this is an eligible agricultural service. The farmer, in turn, allocates the costs of harvesting and transport thus incurred to the operator of the BGP by including them in the production costs of the biomass. Thus, the farmer sells the pre-produced maize chaff, including delivery.

This mode of accounting is more economical for all parties involved but complicates the involved business transactions. In our reference scenario, the BGP's harvest campaign involved 15 farmers and one AC. By the above process, a total of 30 invoices were created by the 17 different managers. However, the relevant data was initially already available in a bundled form within a single document, i.e., the weighing record. The resulting additional bureaucratic effort is justified by the procedure's higher cost-effectiveness but offers significant potential to digitize further and automate these processes, e.g., by tracking the fulfillment of services in near-real-time.

Smart Contracts for Agricultural Logistics In [5], we conceptualized how a digital assistance system could automate this documentation and accounting process.

The article's central idea is to apply smart contracting concepts to digitally capture the contractual service agreements between a logistic service provider and a respective consumer. In this case, an agricultural service provider and the farmer.

Firstly, we considered digital mappings of the logistic service in terms of a *smart contract*. This term refers to a digital representation of a contract that describes the scope and targets of a particular service. It is not necessarily a legal contract in the conventional sense but rather a detailed descriptive model of the service's content covering the involved goods, delivery dates, handling conditions, and other service level agreements. Additionally, it can be augmented with machine-readable rule sets and other forms of business logic that can be used to operate over the contract description. In this way, smart contracts allow to automatically track its contents and proceed its fulfillment by triggering transactions upon certain defined milestones.

Within the article, we detail how smart contracts are closely linked to blockchain technology and argue that blockchains provide tamper-proof data storage an increased level of privacy needed to capture sensitive business-related data. In particular, we use multi-chain structures to interlink multiple, domain-specific chains for the business process, the work process, and the financial transactions, as shown in Figure 2.27.

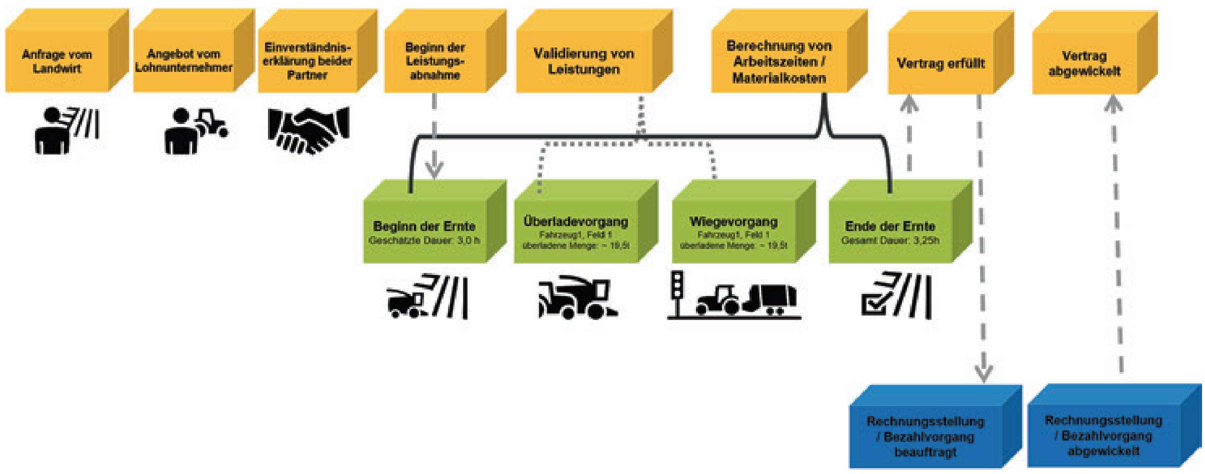


Figure 2.27: Mapping of the forage maize harvest process onto a blockchain

However, the basic principles of smart contracting are technology agnostic and could be implemented on other data representations in combination with suitable security technology instead of a blockchain.

Our work on semantically modeling agricultural logistics processes was at the basis of formulating such contractual models. In general, ontological representations can help to connect process descriptions regarding the work process or business domain. In the included article, however, we did not utilize formal knowledge representations since these are not yet compatible with available blockchain representations. However, an extension would be straightforward for off-chain implementation of the smart contract's business logic.

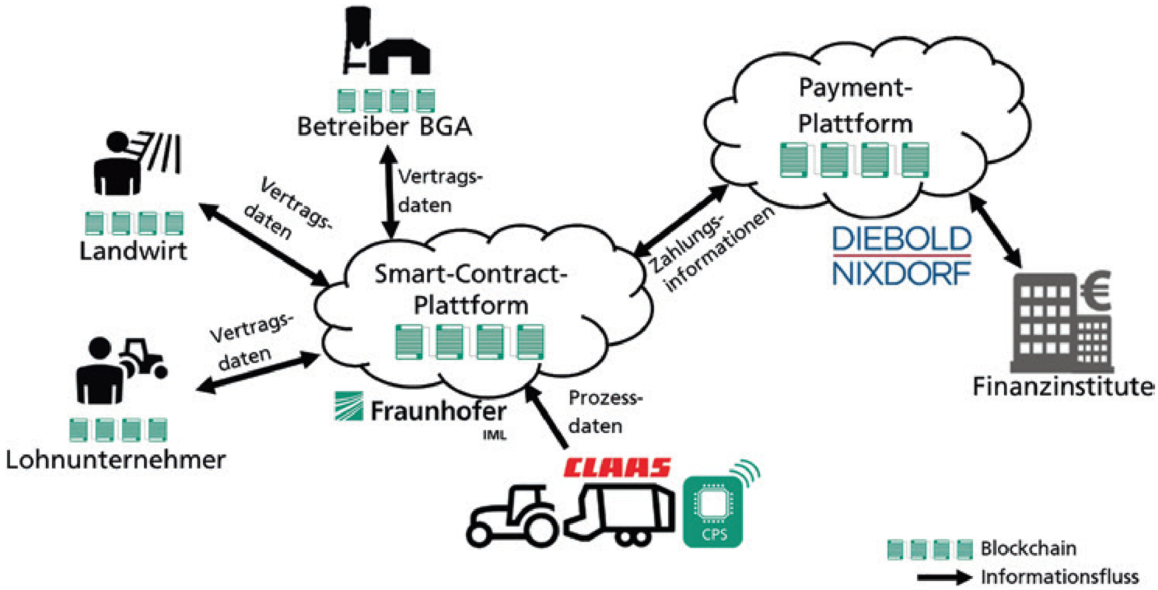


Figure 2.28: The architecture of the proposed smart contracting platform.

Secondly, we propose concepts for a Smart Contract Platform (SCP) that allow us to monitor logistic processes in near-real-time and evaluate the state of a digital service agreement and automatically trigger billing and invoicing transactions after service fulfillment. Its high-level architecture is depicted in Figure 2.28.

The system is designed as a centralized component that realizes two primary functions. On the one hand, it provides interfaces to the customer for business-related activities, i.e., creating, negotiating, and storing digital contracts or service agreements. For example, a web-based interface can allow business partners to negotiate and define the scope of logistic services and agree on the necessary rules and regulations to be monitored during execution. On the other hand, it provides the processing units to evaluate all currently running contracts with the associated rules and regulations, to automatically update the contracts states, trigger transactions in real-time, and inform the business partners about the unfolding service. Once all parties agree upon the contracts, they are automatically managed, checked, and processed based on application-specific business logic. This allows to check services for fulfillment and inform or warn the involved business partners if service contracts are breached.

The SCP must be connected to a system that provides data and knowledge about the ongoing work process to enable these functions. For this, we utilize our process monitoring prototype. It is important to highlight the role of our OPA system in this context. Due to its qualitative and quantitative outputs in a defined semantic representation, it immediately provides the kind of highly aggregated process data that can be used in the SCP to match contractually defined service goals with the currently executed process.

In summary, the online process monitoring capabilities can be used to implement commonly known documentation features in a more timely fashion. Likewise, the usage of semantic models for agricultural logistics processes support the digitization of related work process and in combination with smart contracting technology can automate the tracking of service fulfillment and remove unnecessary bureaucracy. This allows agricultural managers to focus more on operations management and business-related decision-making.

2.6.2 Towards Decentralized Planning & Control

Another potential application of our process monitoring solution is in planning and control systems of agricultural logistics. As discussed by Bochtis et al., activity monitoring modules are essential for realizing dynamic planning and control. Our OPA system and its grounding functions are exactly such an activity monitoring module.

In the SOFiA project, we worked towards decentralized planning and control solutions, and the included articles [1] and [2] address this subject conceptually. The ideas presented in those articles partially motivated the work on online process monitoring presented in this thesis.

Regarding the control of logistic processes, one of the project's central assumptions was that a decentralized approach is a promising alternative to current approaches. As discussed in Section 2.2.5, currently, most planning and control solutions are based on centralized architectures, in which a central node is tasked with global planning making. In this configuration, the central node alone addresses aggregating data and planning for the entire process chain.

Hence all process participants must be connected in a star-shaped network structure to the central node. All data has to be sent from the process participants to the central node, and all control commands have to be returned to them after each planning and control loop. This architecture is disadvantageous for logistics processes, since these are highly distributed, making constant communication between executing entities and a single decision node difficult.

For agricultural logistic processes, where process participants are placed in rural areas, this is particularly problematic. A lack of infrastructure, such as insufficient mobile phone coverage, makes establishing a consistent, centralized database difficult in these circumstances.

Therefore, we propose investigating more decentralized approaches since they may be better suited to address problems where they arise and thus solve them on site.

Distributed Decision-Making based on Hierarchical Task Decomposition Our decentralized planning and control concepts are based on the idea that decision-making could be delegated to the individual distributed process participants. This would reduce the necessity for gathering data in a central node and resemble the current mode of operational decision-making used by agricultural practitioners but enhanced through digital assistance.

Decision support tools or automated decision-making solutions could be implemented onboard of the agricultural machines directly to bring assistance closer to the machine operators. In this way, one would trade the necessity of communication to a central node against having to implement a higher degree of process understanding onboard of the machines and also introduce M2M communication for controlling cooperative activities.

Such an architecture layout requires that the agricultural logistic processes be segmented into independent sub-processes, which can be worked and decided upon by individual entities or by cooperating entities. Since machines often interact closely during harvesting, it is crucial to divide processes to provide a high degree of independence for the executing machines.

As discussed in Section 2.4.1, we decomposed the process of forage maize harvesting using spatial cues. The resulting hierarchical structure orders the different sub-processes that occur during forage maize harvesting such that every layer provides a set of process steps, which are worked by a specific type of process participant, cf. Figure 2.17. During planning, each process participant would be assigned to work on these process steps in the form of one or many assignments. Each assignment is created by a partial delegation from the process step or assignment in the layer above. For example, a harvesting fleet's harvesting tasks are broken into distinct chopping tasks for one of the fleet's harvesters.

The separation of steps is constructed such that they can be resolved by the respective process participants, without any alignment with other processes of the same kind or within the same layer, at least as long as the process is running within the bounding parameters defined in the assignment. For example, within a harvesting fleet's assignment to harvest a particular field, the order of transport vehicles may be arbitrarily changed and optimized, without any implications to the other fleets operations, as long as the scheduled time-frame for the field's harvest with respect to the global contract is not broken.

In this manner, the model segments the processes from top to bottom. Likewise, during the (re-)planning phase, the task assignment follows a top to bottom delegation process. For the alignment of interdependent process steps, status and result data aggregation are reported from bottom to top. However, there must be some form of state propagation when processes need to be monitored and controlled across multiple layers of the model's hierarchy. Its details are dependent on the process type and control approach. Hence the demand for a centralized alignment within the decentralized network may vary. Some tasks, e.g., a single transport run from field to farm, may require no alignment with other partners within the distributed network at all. In contrast, balancing the material flow between fields and farms may require the alignment of multiple fleets.

At the top node of the model, a contract between a farmer and contractor is defined, which denotes the scope and contractual parameters of a joint harvesting campaign. It encapsulates the business-related negotiations and resource alignments. As a decision task, it is mainly subject to human decision-making by the respective stakeholders. The level below this node is concerned with the harvesting fleets and compaction fleets, which work within the given campaign. Here, all the individual fleets' assignments are formulated based on the global contract and then delegated.

Since a fleet consists of multiple machines, it is represented by multiple decision nodes in the distributed decision network. One node is selected as the fleet leader to make decisions on behalf of the entire fleet. The leader might be chosen due to various reasons. It may be the node coming in contact with all relevant decision partners on a regular basis, or it may have the best access to the data required to make a decision, e.g., due to unique communication capabilities.

Our model also exploits that forage maize harvesting is a spatially distributed process. The details of the processes can be controlled locally, and only some high-level alignments need to be done globally. The local sub-processes of a harvest in the field are partially independent of those related to storage and compaction at the silo. The detailed procedures of these spatially remote operations need not be synchronized in-depth but be aligned on relevant KPIs, e.g., the flow of transported goods, and roughly follow the same schedule. Therefore, they can be primarily coordinated locally within the respective fleets or onboard a single machine. If a harmonization between the processes is required, e.g., balancing the amount of transported goods, this can be done on demand by synchronizing the respective fleet assignments via a mutual control loop. In this step, both fleets would try to align their behavior with the parametrized boundaries defined in the contractual background. If the negotiated changes to the plan do not influence the contract boundaries or the detailed plans within the fleet's sub-processes, no decision propagation into the distributed network is necessary. If, however, the contractual boundaries can not be respected during a bilateral re-planning, the decision may be escalated to the contract layer, where the business stakeholders ultimately resolve it. Likewise, any implications on the sub-process will be propagated down to the individual process participants, if necessary. There it will trigger a re-planning if the boundary parameters have changed significantly. This does not necessarily affect all participants within the newly synchronized fleets. If, for example, a shortage of material at the silo's forecourt shall be remedied by speeding up the transport volume, the goal values of shorter transport cycle times and increased average speeds, only be distributed to the transport vehicles. This does neither affect the harvesting strategy of the fleet's harvesters nor the compaction vehicles' schedules.

As discussed in the included articles, we are aware that this approach might not yield mathematically optimal solutions since it can not be guaranteed that all required information is available at all times in each node of the distributed decision network. Likewise, we recognize the added complexity of decision brokerage between the multiple decision-makers without having an immediate solution for this problem. However, we propose further research on decentralized approaches. Despite the increased complexity and potential limitations, a positive trade-off might be made because it enables dynamic control even during communication breakdowns. Furthermore, a response on-site allows for a more timely control response, which is beneficial during time-critical operations.

Process Monitoring in Decentralized Architectures Through the decomposition of tasks and by placing the responsibility of decision-making onto the process participants, all onboard decision nodes need to know about their assignments and current progress.

It also follows that every participant has a demand for state knowledge and planning capabilities. Hence process monitoring capabilities on every decision node is a pre-requisite for our concept. It is required that all digitized logistic entities understand their role within the logistic process and can infer their state and progress within the logistical network. Within a decentralized network, this inference has to be *online* and *onboard*. By online, we mean that the inference is quick enough to allow for an immediate reactive response to a sub-optimal logistical situation, but ideally is even prospective, in a sense that an arising shortage is detected early enough to allow for a pro-active response. By onboard, we mean that the inference mechanism must be situated in or at the logistic entity.

This thesis focuses on addressing an online solution to process monitoring, disregarding the onboard requirement. This deviation from the SOFiA project's targets was intentional. To infer and reason about a logistic entity's current state and progress is an essential prerequisite in any form of a digital assistance system for logistic planning and decision-making, regardless of the architectural layout. The application in a decentralized context is just a matter of deployment and technical feasibility for the core principles of grounding process knowledge.

Regarding decentralized decision-making, solving the state inference and process monitoring problem takes precedent over the particular concerns of a decentralized decision approach. Based on this insight, we for now omitted a direct application in process planning and decision making.

However, our prototypical implementation of online process monitoring capabilities can be utilized in a centralized and decentralized deployment. For the sake of simplicity, we opted to discuss all results in the context of a centralized architecture since this would also be the first form of deployment from which a decentralized version could be derived in future work.

In principle, our OPA system can be applied onboard of single machines, which would allow for individual decision nodes to determine their state themselves. At least for single-machine tasks, this application is straightforward. This opens up further research into how to plan for the appropriate behavior of the individual machine in a decentralized manner.

Our prototypical system is also favorable in cases where communication with partnering process participants is required for mutual coordination of activities. The communication between process participants can be based on highly aggregated data, such that only the relevant information for decision-making is shared. This alleviates the necessity of managing synchronized data set within the central node and is beneficial given the often constrained communication capabilities in agricultural environments.

In summary, the project goals of the SOFiA project were aspirational for this thesis, and the results of our presented work can contribute towards future work on decentralized planning and control approaches for agricultural logistics systems.

Chapter 3

Summary & Outlook

3.1 Summary

This thesis concerned the analysis of ongoing agricultural logistic processes to derive explicit process information from machine telemetry to better inform agricultural practitioners during operational decision-making, and to potentially enable digital assistance systems for planning and control to become more dynamic in the future.

We based our work on a review of current digital assistance systems for harvesting processes. We found that planning systems for agricultural processes are well understood but hardly applied at process runtime at all, because there are no tools for providing state information and feedback on preplanned targets, i.e., no digital solutions to provide online process monitoring.

Hence, we proposed concepts for process monitoring based on the observation that agricultural logistic processes, e.g., forage maize harvesting, are inherently spatially distributed, and thus can be enabled through spatial analysis. We also argued that process monitoring should be implemented based on a semantic model of the agricultural process and its involved entities to enable automated reasoning and inference in conjunction with additional background knowledge.

We found in our review that all the pre-requisites for such spatio-semantic reasoning are readily available. Agricultural machines already produce geo-referenced streams of telemetry data due to the ubiquitous use of GNSS localization. It is also common to represent agricultural environments in terms of spatial maps. Hence the basis for quantitative, geometric analysis is given and allowed us to transition to spatio-semantic reasoning by integrating formal knowledge representations into the existing data analytics solutions. Likewise, there is initial work on semantic modeling in the agricultural domain, including some ontological models regarding harvesting processes and their sub-tasks, which we used for reference in our models.

We developed the SEMAP system to provide integrated spatio-semantic representation and reasoning. It is a semantic mapping framework that allows representing semantic maps and deriving explicit knowledge about the spatial relations between entities and their environment. Architecturally, it combines spatial databases with knowledge bases to connect dedicated spatial analysis and querying with formal reasoning and inference, respectively. We also utilized formal semantics since machine-readable knowledge representations enable ontological and rule-based reasoning over the information stored in the semantic maps. In particular, we used semantic web technologies because their underlying ontological representations allow us to combine domain-independent and domain-dependent models. Thus SEMAP itself is domain-agnostic but can easily be applied in different application contexts, e.g., robotics or agriculture.

We utilize SEMAP to implement our concepts for online process monitoring prototypically. We first created a semantic model of the agricultural environments and machinery. We also modeled concepts and relations to describe the process states and events occurring during forage maize harvesting. We then demonstrate how to apply spatial analysis and rule-based reasoning within SEMAP to learn about ongoing agricultural processes in this semantic model. We focused our work on implementing two types of grounding functions that can derive qualitative and quantitative from the telemetry data of agricultural machines. Our qualitative classification functions can detect the process states in the transport logistics of forage maize harvesting in an event-based fashion. Building upon the classification results, we also demonstrated measuring process times as an example of measuring process-related key performance indicators. We further demonstrated how to implement automated process documentation and accounting as a customer-facing feature related to business-related tasks of managing a harvesting operation.

We also discussed the ability to monitor ongoing agricultural processes in the context of dedicated online process analytics systems. We intend this new category of analytics system to be an extension of the system-of-systems context of digital assistance solutions for agriculture, that complements systems for planning and execution control to enable dynamic re-planning and adaptive process optimization.

In summary, we were able to answer the posed research questions regarding semantic mapping and digital agriculture. We showed how to combine spatial analysis and semantic reasoning techniques for grounding new knowledge from a multi-modal environment representation. We implemented our concepts in a domain-agnostic form easily extended into specific domain applications using semantic technologies. We utilized it to derive process information about ongoing agricultural logistic processes through spatio-semantic reasoning techniques. We demonstrated applications that benefit the monitoring and documentation of ongoing agricultural operations. Finally, we discussed the impact of such capabilities in the context of planning and control systems for agricultural logistic processes.

3.2 Outlook

Several subjects for future work can be derived based on the results of this thesis.

Experimental Evaluation & Ground Truth Data Acquisition Our experiments and applications showed the conceptual feasibility of spatio-semantic reasoning for monitoring ongoing agricultural logistic processes. However, it is important to note that our proof-of-concept was conducted in the lab. Even though we worked with data recorded in a real-world harvesting campaign, the deployment of online process monitoring capabilities in a real-world application would undoubtedly bring additional unexpected disturbances, which would probably not be covered given our current feature set. Additional work on the proposed grounding functions is expected to make it robust enough for real-world application.

We aim to conduct an experimental evaluation in multiple real-world harvesting campaigns within future work to identify the limits of our concepts and prototypical implementation.

Related to this, we realize some challenges regarding the collection of suitable ground truth data. Ideally, one would be able to compare the results of an OPA system against a set of high-quality process documentation that precisely annotates which process state and events held during operations. However, gathering such data is laborious and inherently subject to the rules applied for labeling. Defining these rules is somewhat subjective and open to debate, as is the implementation of our grounding rules. We believe that additional discourse with practitioners will be most effective in assessing our method's applicability and closely matching the customers' expectations regarding the classification of a harvesting process within future implementations of online process monitoring functions.

Explicit Temporal Representation & Reasoning This work emphasized that semantic representations have to capture the temporal aspects of the monitored process and, therefore, developed an event-based process model. However, our semantic mapping framework SEMAP currently does not directly utilize any explicit models of time. This hindered us from implementing state comparisons and event generation directly in SEMAP, such that we had to implement additional state tracking. This is a fundamental conceptual limitation in our work.

Future implementations should natively include temporal representations and dedicated temporal reasoning capabilities to complement our current approach of spatio-semantic reasoning. Toward these subjects, a large body of conceptual work is available in the scientific literature. There are also tools in the semantic (web) technology stack to base future work on.

Performance Improvements & Stream Processing We also emphasized the importance of near-real-time processing capabilities for OPA systems. However, we realized that through the dual database setup and hybrid query system, SEMAP is technically constrained in the amount of spatio-semantic queries that can be performed in short succession. Especially, running inferences over the knowledge bases quickly becomes computationally expensive in our agricultural application, where telemetry updates are provided every second. This leads to a mismatch in query and insertion times required by the spatial database and knowledge base, respectively. While the spatial database is quick to update and retrieve from, the knowledge base requires time to update and process new knowledge.

The near-real-time application of SEMAP in agriculture indeed introduced a dynamism that was not fully anticipated in our original implementation. Though we did not reach any performance bottlenecks in our lab experiments, scaling our prototype to monitor multiple harvesting processes in parallel would certainly not be feasible given the current setup.

We proposed to consider general stream processing technology for scaling our spatio-semantic approach to the demand of supporting real-world harvesting processes, e.g., when being deployed in the context of a commercial digital assistance system. Likewise, dedicated stream reasoning technology could be of interest for closely integrating rule-based reasoning directly into the grounding process. It should be possible to derive most of the spatial predicates that we are currently grounding through the interaction of spatial database and knowledge base while still in stream processing. Hence, a processing and reasoning system operating on data streams could address the task of symbol grounding such that already aggregated facts could be inserted into the hybrid representation for persistence and asynchronous inference.

Development of Additional OPS Features We focused on monitoring forage maize harvesting processes and covered only a limited scope of processes states, types, and events. Certainly, our method can be transferred and extended towards other agricultural processes. Here agricultural logistic processes with inverse material flow, e.g., slurry operations, lend themselves as good candidates based on our current results. However, there seem to be interesting topics to explore when integrating OPA systems into the control architecture of agricultural robots.

Towards Planning, Control & Decision Assistance Solutions Another definite subject for future work is integrating our OPA system with dedicated planning and control architectures for agricultural processes. This thesis work was motivated by the observation that dynamic planning and control without dedicated monitoring is futile. Now that we are capable of deriving knowledge about process states and measuring related KPIs, methods of coupling monitoring and re-entrance-capable planning systems with each other must be developed.

We assume that the operationalization of process monitoring in centralized planning systems is currently more within state of the art and should be explored first to leverage the available potential for optimization. However, we like to point to the potential issues of communication breakdowns in agricultural environments. Our approach relies on analyzing a continuous data stream, such that if there is a data outage, misclassification and tracking issues are to be expected.

Future work has to propose concepts for making the OPS functions robust. Especially recovering states and processing batches of old data to catch up with near-real-time processing are relevant. Having access to explicit temporal semantics should aid in managing such situations. Another avenue is deploying OPA components on board to create a distributed monitoring system.

Therefore, we also strongly encourage following up on decentralized concepts for planning and controlling agricultural processes. Adding concepts from multi-agent systems could aid in structuring and resolving the challenges that come with multiple deciders in a decentralized decision system, as envisioned by the SOFiA project. In this line of future work, smart contracting could also be included to document acts of distributed (re)-negotiations and joint decision-making across digitally augmented human or fully autonomous virtual agents.

Privacy Concerns Finally, there are legal concerns to be addressed regarding the subject of process monitoring. As long as agricultural work is conducted by humans operating agricultural machines, process monitoring always entails operator monitoring.

The privacy concerns of machine operators need to be considered and taken seriously when the subject of online process monitoring matures into commercial application.

Part II

Scientific Publications

List of Attached Papers

[1] H. Deeken, T. Steckel, and M. Witthaut. Verbesserung logistischer Prozesse in der Landwirtschaft durch Betrachtung von Maschinen als smarte Objekte. In *LAND.TECHNIK 2016: Das Forum für agrartechnische Innovationen*, pages 397–404. VDI, VDI Verlag, 2016.

[2] H. Deeken, F. Krampe, and T. Steckel. Verbesserung logistischer Prozesse durch Dezentralisierung von Entscheidungen. In *Referate der 37. GIL-Jahrestagung - Digitale Transformation - Wege in eine zukunftsfähige Landwirtschaft*, Lecture Notes in Informatics (LNI), pages 41–44. Gesellschaft für Informatik in der Land-, Forst- und Ernährungswirtschaft (GIL), Gesellschaft für Informatik, 2017.

[3] H. Deeken, T. Wiemann, and J. Hertzberg. Grounding Semantic Maps in Spatial Databases. *Robotics and Autonomous Systems*, 105:146–165, 2018.

[4] H. Deeken, T. Wiemann, and J. Hertzberg. A Spatio-semantic Approach to Reasoning about Agricultural Processes. *Applied Intelligence*, 49(11):3821–3833, 2019.

[5] D. Sparer, H. Deeken, B. Künsting, and P. Sprenger. Smart Contracts und Smart Payment im Farming 4.0. In *Digitale Dienstleistungsinnovationen: Smart Services agil und kundenorientiert entwickeln*, pages 445–471. Springer, 2019.

Statement on Co-Authorships

The publications regarding the conceptual ideas behind the SOFiA project originated in discussions with colleagues at CLAAS and with the partners within the project consortium. I authored the majority of the two included articles with feedback and revisions by the co-authors. In the case of [1], corrections and additions were made by Thilo Steckel and Markus Witthaut. In the case of [2], Thilo Steckel and Florian Krampe were involved, respectively.

Improvements to Agricultural Logistics by Digitalization and Decentralization The publications regarding the conceptual ideas behind the SOFiA project originated in discussions with colleagues at CLAAS and with the partners within the project consortium. I authored the majority of the two included articles with feedback and revisions by the co-authors. In the case of [1], corrections and additions were made by Thilo Steckel and Markus Witthaut. In the case of [2], Thilo Steckel and Florian Krampe were involved, respectively.

Grounding Semantic Maps in Spatial Databases The initial work on the SEMAP framework originated in my Master's thesis. It was substantially extended and revised during my Ph.D. project, which resulted in the included publication [3]. Thomas Wiemann, who supervised both theses, proposed using spatial databases to manage and distribute spatial maps to mobile robots. I conceptualized and implemented his initial intentions to use the built-in operators of PostGIS to implement sub-map queries and the relational components of the database to store semantic annotations. The use of qualitative spatial operators and formal semantics for spatial reasoning were developed by me. The concepts for integrating a dedicated knowledge base component to handle factual and domain knowledge were jointly developed. The required research, ontological modeling, and technical implementation were entirely done by myself. The included publication [3] is a significant revision of my Master's thesis and incorporates results of previous and subsequent publications on SEMAP [77–79]. The writing was done primarily by me with significant input from Thomas. During the entire work on the SEMAP framework, Joachim Hertzberg provided continuous conceptual and editorial feedback.

At time of writing, this article was cited 15 times according to Google Scholar.

A Spatio-Semantic Approach to Reasoning about Agricultural Processes This publication was based on the idea of applying concepts of semantic mapping onto the agricultural domain by using the SEMAP framework. It summarizes my contributions to semantic mapping and the online process monitoring of agricultural logistic processes. The research, ontological modeling, and technical implementation were entirely done by myself. As were most of the conceptual work, to which Thomas Wiemann and Joachim Hertzberg provided valuable feedback.

The work was originally published in the context of the *31st International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems* [79]. The included publication [4] is a revised and updated version published in a special issue of the Applied Intelligence Journal. We were kindly invited to contribute to based on our conference paper.

At time of writing, the included article was cited 2 times, and the original conference article 6 times, according to Google Scholar.

Smart Contracts und Smart Payment im Farming 4.0 This article [5] was a collaboration within the SOFiA project and summarized the project's results of the smart finance aspect. Dominik Sparer and Philipp Sprenger of the Institut für Materialfluss und Logistik, who coordinated this topic within the project, lead the writing and editing process. All authors contributed to the text in equal parts. Sparer and Sprenger contributed the sections on applying blockchains and smart contracts to logistic processes and the technical considerations on using multi chains. The description of the agricultural use case and the application of multi chains onto the forage harvesting process were conceptualized and authored by me. Björn Künsting of Diebold Nixdorf contributed the sections on the automated payment and billing system. The article was published in the context of the *BMBF-Förderlinie Dienstleistungsinnovationen durch Digitalisierung* and – for what it's worth – won the Digivation's Best Paper Award.

At time of writing, the included article was cited once, according to Google Scholar.

Chapter 4

Improvements to Agricultural Logistics by Digitalization

4.1 Verbesserung logistischer Prozesse in der Landwirtschaft durch Betrachtung von Maschinen als smarte Objekte

Henning Deeken CLAAS E-Systems KGaA & Co KG, Gütersloh, Germany
Markus Witthaut Fraunhofer-Institut für Materialfluss und Logistik, Dortmund, Germany
Thilo Steckel CLAAS E-Systems KGaA & Co KG, Gütersloh, Germany

Abstract

Logistische Prozesse in der Landwirtschaft sind durch schwer vorhersagbare Randbedingungen häufig von Störungen und Kapazitätsengpässen gekennzeichnet. Eine dezentrale Vernetzung und Entscheidungsfindung erscheint viel versprechend, durch frühzeitige Erkennung und Behandlung von Problemen vor Ort diese Situation zu verbessern. Im Projekt SOFiA wird hierzu die Verknüpfung semantischer Prozessmodellierung in Kombination mit ereignis-diskreter Simulation zur dezentralen Entscheidungsunterstützung erforscht. Im Folgenden werden die dem Projekt zugrundeliegenden Konzepte am Anwendungsfall der Silomaisernte vorgestellt.

4.1.1 Einleitung

Logistische Prozesse sind in fast jedem Fall Bestandteil von landwirtschaftlichen Wertschöpfungsketten. In der Praxis erweist sich der Transport oder Umschlag von Gütern als Herausforderung, da neben einer Vielzahl von involvierten Maschinen und betrieblichen Akteuren, auch eine hoch dynamische Umgebung den Ablauf der Logistik beeinflusst. Die planbasierte Steuerung komplexer Lieferkette spielt in der Landwirtschaft noch eine untergeordnete Rolle. Wichtige logistische Entscheidungen, z.B. wie viele Ernte- und Transportfahrzeuge in der Ernte eingesetzt werden und wie wann diese an Feld und Lagerstätte bereitstehen müssen, wird überwiegend informell geplant. Gelegentlich werden hierfür Dispositionssysteme landwirtschaftlicher oder domänen-unspezifischer Prägung eingesetzt. Die anschließende Steuerung der Prozessausführung erfolgt überwiegend informell (mündlich, fernmündlich, per Auftragszettel). Schlägt der Plan zur Ausführung fehl, wird für die nötige Reaktion in der Regel auf die Erfahrung der Beteiligten gebaut. Selten werden Entscheidungen auf einer klaren quantitativen Faktenbasis abstützt, da hierfür adäquate Methoden und Technologien kaum vorhanden sind. So kommt es in der Ernte oft zu logistischen Engpässen und Störungen im Prozessablauf, z.B. wenn ein Feldhäcksler auf ein Transportfahrzeug zur Aufnahme des Ernteguts warten muss. Insbesondere Wartezeiten und suboptimale Arbeitspunkte reduzieren die Auslastung der Maschinen deutlich und verschlechtern den wirtschaftlichen Erfolg. Die zunehmende Ausstattung landwirtschaftlicher Arbeitsmaschinen mit informations-verarbeitenden Systemen, sowie die drahtlose Vernetzung der Maschinen untereinander, kann jedoch zur Verbesserung transportlogistischer Aufgaben genutzt werden, wenn eine entsprechende Digitalisierung der Prozesse umgesetzt wird und die Prozessteilnehmer mit geeigneten Plan- und Steuerungssystemen ausgestattet werden.

4.1.2 Projekt SOFiA

Im Folgenden werden erste Ergebnisse des vom BMBF geförderten Projektes Prozessinnovation in Planung und Steuerung von Wertschöpfungsnetzwerken durch Integration von Smart Objects und Smart Finance Ansätzen (SOFiA) vorgestellt, das die Digitalisierung und Dezentralisierung von Planungs- und Steuerungsprozessen in logistischen Wertschöpfungsketten, sowie deren finanztechnischen Abwicklung zum Ziel hat.

In dem Projekt arbeiten das Fraunhofer Institut für Materialfluss und Logistik, der Landmaschinenhersteller CLAAS, der Informationstechnikkonzern Diebold Nixdorf und der Logistikdienstleister EKOL zusammen. Im Projekt SOFiA wird ein verteiltes Netzwerk aus Smart Objects (SO) erzeugt, das Maschinen, Sensoren und Menschen dezentral vernetzt und ermöglicht, Prozesse auf quantitativer Basis unmittelbar auf der Prozessebene zu steuern. Hierzu werden die beteiligten Maschinen dahingehend digitalisiert und vernetzt, dass sie ihre Rolle im Prozess zu verstehen, ihren jeweils eigenen Prozessstatus zu ermitteln, um ihn mit anderen Prozessteilnehmern auszutauschen und in gegenseitiger Abstimmung angemessene Verhaltensweisen zu bestimmen. Der vorliegende Artikel beschreibt das Architekturkonzept eines Smart Objects, sowie den in SOFiA verfolgten dezentralen Steuerungsansatz bestehend aus der Kombination von verteilter Datenhaltung in einem dezentralen Netzwerk aus Maschinen und der Verwendung ereignis-diskreter Simulation und Expertenwissen zur lokalen Entscheidungsfindung. Illustriert wird der Ansatz am Beispiel der Logistik der Silomaisernte. Ebenso wird im Projekt die digitale Abrechnung von logistischen Dienstleistungen durch die Projektpartner Diebold Nixdorf und das Fraunhofer IML untersucht.

4.1.3 Dezentrale Prozesssteuerung

In der Vergangenheit ist die Planung logistischer Wertschöpfungsketten (engl. suppy chains) oft über zentrale Steuerungsansätze realisiert worden. Diese Ansätze bauen auf einen zentralen Knotenpunkt an den alle weiteren Prozessteilnehmer (die ausführenden Maschinen, aber auch externe Datenquellen) in einem sternförmigen Netzaufbau verbunden sind. Der zentrale Knoten übernimmt die Aufgabe Daten zu aggregieren und die Planung für die gesamte Prozesskette durchzuführen. Zur Vorplanung erweist sich dies als sinnvoll, da bei vollständig vorliegenden

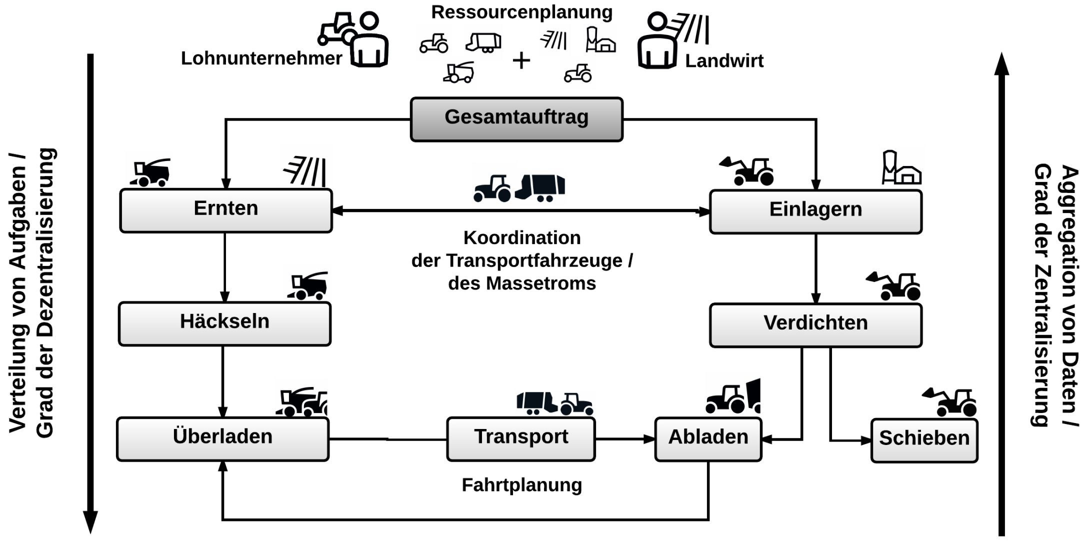


Figure 4.1: Zur dezentralen Steuerung wird der Ernteprozess in unabhängige Teilprozesse segmentiert, welche die Maschinen als Aufträge planen und ausführen.

Informationen über den Prozess mathematisch optimale Lösungen gefunden werden können. Dies ist jedoch nur dann in der für praktische Anwendungen erforderlichen Planungsgüte möglich, wenn sämtliche für die Planung relevanten Zielgrößen und Randbedingungen der Supply Chain in einem quantitativen Modell abgebildet werden [224].

Zur Steuerung von verteilten Prozessen, wie in der Landwirtschaft, ist die Verwendung eines zentralen Steuerungssystems von Nachteil: Zuerst müssen alle Prozessinformationen von den Teilnehmern in das zentrale System gesandt werden, und ebenso alle Steuerbefehlen nach der Planung an diese zurück gegeben werden. Insbesondere bei räumlicher Trennung der Prozessteilnehmer in Kombination mit infrastrukturellen Problemen, wie mangelnder Mobilfunkabdeckung, kostet die zentrale Planung mehr Zeit als eine adäquate, zeitnahe Reaktion erlaubt. Dynamische Anpassungen zur Laufzeit des Prozesses über zentrale Optimierung sind somit in verteilten Prozessen selten möglich.

Dezentrale Ansätze hingegen beruhen auf der Idee, die Entscheidungsfindung auf die einzelnen Prozessteilnehmer zu verteilen und in räumlich eingegrenzten Teilprozessen durch direkte End-zu-End Kommunikation genügend Informationen zusammen zu tragen, um lokale Prozessoptimierung zu betreiben. Entscheidungen sollen unmittelbar dort getroffen werden, wo ein Problem auftritt. Damit eine dezentraler Prozesssteuerung gelingt, ist es nötig den betrachteten Prozess in möglichst unabhängige Teilschritte zu segmentieren. Bei der Silomaisernte ist dies möglich, da die ortsgebundenen Unterprozesse der Ernte im Feld teilweise unabhängig von denen der Einlagerung am Silo sind und somit auch größtenteils lokal kommuniziert und koordiniert werden können. Es wird angenommen, dass jene Prozesse, in denen mehrere Maschinen in räumlicher Nähe interagieren, auch diejenigen sind, die schnelle Reaktionen erfordern und auch auf den Austausch größere Mengen von Daten angewiesen sind.

Die Koordination des Überladevorgangs vom Feldhäcksler auf ein Transportfahrzeug erfordert eine zeitnahe, durchgehende Kommunikation zwischen beiden Maschinen zur Koordination von Fahrtrouten oder der Steuerung der Überladeautomatik und zum Austausch der übergeladenen Erntemenge. Dieser Kommunikationsbedarf ist vollständig lokal, es kann also direkte M2M Kommunikation mit hoher Konnektivität genutzt werden um die relevanten Informationen für die Planung direkt vor Ort zu aggregieren. Im Gegensatz dazu haben räumlich entfernte, globale Steuerungsprobleme deutlich mehr zeitlichen Spielraum zur Reaktion auf unvorhergesehene Ereignisse und lassen sich zudem meist auf kompakte Statusinformationen reduzieren, sodass auch bei eingeschränkter Konnektivität noch eine robuste Prozessteuerung über mehrere Prozessteilnehmer hinweg realisiert werden kann. Ein Beispiel ist die globale Koordination von ein- und ausgehendem Massestrom im Ernteprozess und das Scheduling der Transportfahrzeuge mit entsprechenden Verteilung der Erntemasse auf deren Kapazitäten. Dieses globale Steuerungsproblem lässt sich auf den Austausch leichtgewichtiger Informationen, wie z.B. der geplanten und tatsächlichen Ankunftszeiten der Transportfahrzeuge, reduzieren und wird im dezentralen Ansatz auf einer oder mehreren Maschinen, anstatt in einer zentralen Instanz geplant. Eine optimale Lösung im mathematischen Sinne lässt sich mit dezentralen Ansätzen zwar nicht immer finden, aber im Hinblick auf die praktischen Anforderungen der Silomaisernte verspricht ein dezentraler Ansatz mit lokaler Optimierung des Prozessablaufs eine wesentlich robustere Steuerung, da der Kommunikationsaufwand und die Latenzen eines zentralen Ansatzes wegfallen.

4.1.4 Smart Objects

Um die Planung und Ausführung der Teilprozesse maschinell auf den Maschinen durchzuführen, werden die am Prozess beteiligen Maschinen (Selbstfahrende Feldhäcksler, Transport- und Verdichtungsfahrzeuge) hard- und softwareseitig so weiterentwickelt, dass sie als Smart Objects ihre Rolle und Handlungen im Prozess verstehen und planen können. Weitere Datenquellen (z.B. Sensoren im Feld oder externer Wetterdienste), die Prozessverantwortlichen (Landwirte und Lohnunternehmer), sowie die Maschinenbediener werden über entsprechende mobile Endgeräte (z.B. Smartphones) eingebunden und ebenfalls als SOs modelliert. Letztere bekommen die zur Prozessoptimierung ermittelten Handlungen, z.B. die vorgeschlagenen Anpassung der Transportgeschwindigkeit oder Änderung des Durchsatzes in den Teilprozessen Ernte und Verdichtung oder die Hinzunahme weiterer Transportfahrzeuge über ein Assistenzsystem vorgeschlagen.

Die Modellierung der Smart Objects folgt dem Agentenansatz (vgl. [92]) und der Idee, dass sich die Teilprozesse der Silomaisernte als Aufträge beschreiben lassen und deren Ausführung von den SOs untereinander verhandelt wird. Hierzu werden die Aufgaben mit Zielvorgaben (wie verfügbare Zeit, erlaubte Kosten oder angestrebte Qualität) versehen und hierarchisch organisiert, sodass die Erfüllung der Ziele von oben nach unten delegiert wird und die Informationen über die Ausführung von unten nach oben aggregiert werden (Abbildung 4.1). Zur Ausführung wird ein Soll/Ist-Vergleich von Prozesszuständen und Zielvorgaben durchgeführt um zu ermitteln, ob Steuerungsbedarfe bestehen, und wenn ja, ob diese lokal konsolidiert werden können oder ob eine Abstimmung mit höheren Auftragsinstanzen nötig ist. Diese Entscheidungen werden dezentral auf den Maschinen getroffen, die die Planungshoheit über den jeweiligen Auftrag halten. Hierzu wird mit den betreffenden Partnern kommuniziert. Die dezentrale Steuerung des Prozesses kann also als Multi-Agentensystem aufgefasst werden.

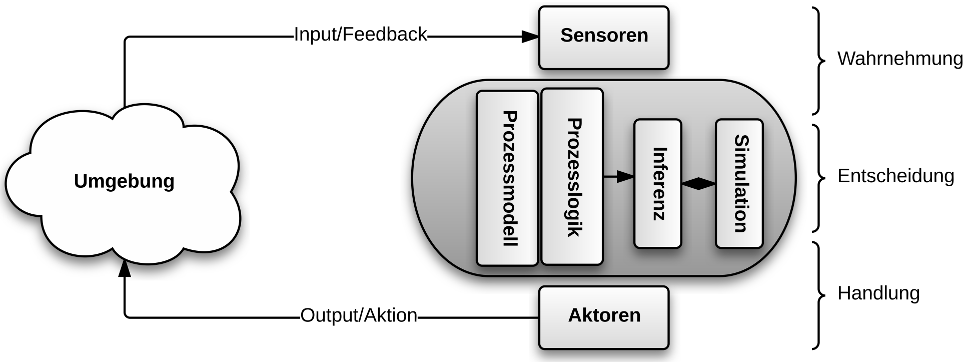


Figure 4.2: Smart Objects pflegen ein semantisches Prozessmodell und extrahieren mittels regelbasierter Logik verschiedene Ernteszenarien, die zur Entscheidungsfindung simuliert und anschließend durch die Prozesslogik bewertet werden.

4.1.5 Architekturkonzept

Die im Projekt konzipierte Architektur der Smart Objects (Abbildung 4.2) sieht vor, dass in einem Prozessmodell der interne Zustand des Agenten, sowie sein Wissen über die andere Agenten, deren Aufträge und die Umwelt abbildet werden. Hierzu pflegt jedes SO eine Wissensbasis, die aus der internen Sensorik des SOs und externen Quellen gespeist wird. Zur Modellierung und Repräsentation von Wissen wird auf Semantic Web Standards (RDF, OWL) [28] zurückgegriffen und Regelsprachen und Inferenzmechanismen (SWRL) verwendet, um regelbasiertes Schließen auf dem Modell zu ermöglichen. Die aus verschiedenen Regelsätzen zusammengesetzte Prozesslogik erfüllt hierbei eine Reihe von Aufgaben: So wird regelbasiert auf den Prozessstatus des Smart Objects geschlossen, indem Maschinendaten und Umgebungsdaten fusioniert werden (z.B. zur Verortung der Maschinen über den Abgleich von GPS Position und Schlagkartei). Ob eine Maschine ihren Auftrag erfüllt oder nicht, lässt sich ebenfalls per Regelwerk prüfen. Die Planung neuer Aktionen lässt sich über Inferenz alleine nicht realisieren, hierzu wird eine ereignisdiskrete Simulation verschiedener logistischer Szenarien durchgeführt. Um geeignete Szenarien zu ermitteln, wird landwirtschaftliches Expertenwissen in Form von Regeln implementiert und genutzt, um zu bestimmen, welche Erntestrategien simuliert werden und um eine geeignete Lösung zu finden. Über semantische Anfragesprachen (SPARQL) können hierzu die entsprechenden Daten gezielt aus dem Prozessmodell extrahiert und an den Simulator übergeben werden. Als Simulator wird das am IML entwickelte OTD-NET Framework verwendet und erweitert [140].

Die hier präsentierten Ansätze einer dezentralen Prozesssteuerung werden im weiteren Fortschreiten des Projekts SOFiA erprobt, erweitert und daraufhin bewertet, ob die erhofften Verbesserungspotenziale für die Planung und Steuerung logistischer Prozesse in der Maisernte erzielt werden können.

4.2 Verbesserung logistischer Prozesse durch Dezentralisierung von Entscheidungen

Henning Deeken CLAAS E-Systems KGaA & Co KG, Gütersloh, Germany
Florian Krampe CLAAS E-Systems KGaA & Co KG, Gütersloh, Germany
Thilo Steckel CLAAS E-Systems KGaA & Co KG, Gütersloh, Germany

Abstract

Die Planung und Steuerung von logistischen Ketten erweist sich in der Landwirtschaft oft als Herausforderung, da unvermeidbare Störungen im Ablauf und Kapazitätsengpässe auftreten. Diese adäquat zu behandeln bedarf einer engen Abstimmung alle Prozessteilnehmer, eine Aufgabe der zentrale Planungsarchitekturen oft nicht gewachsen sind, da es an den geeigneten Kommunikationskanälen von und zur Prozessebene mangelt. Die dezentrale Vernetzung von Maschinen und eine verteilte Entscheidungsfindung auf Prozessebene scheinen viel versprechende Ansätze, um Probleme frühzeitig am Ort des Entstehens zu erkennen und zu behandeln. Das Projekt SOFiA erforscht Verfahren zur dezentralen Entscheidungsunterstützung in logistischen Netzwerken mithilfe sogenannter Smart Objects. Die dem Projekt zugrundeliegenden Ideen werden im Folgenden am Anwendungsfall der Silomaisernte vorgestellt.

4.2.1 Einleitung

Die meisten landwirtschaftlichen Verfahren beinhalten logistische Unterprozesse, wie zum Beispiel Materialflüsse vom Feld zum Lager (Ernte), sowie vom Lager zum Feld (z.B. Düngung, Pflanzenschutz). Durch eine Vielzahl von involvierten Maschinen und betrieblichen Akteuren in Kombination mit einer hoch dynamischen Umgebung kommt es jedoch regelmäßig zu Störungen beim Transport und Umschlag von landwirtschaftlichen Gütern. Zur Reaktion auf diese Störungen wird in der Praxis oft auf die Erfahrung der involvierten Personen gesetzt, welche durch informelle Kommunikation (per Betriebs- oder Mobilfunk) und auf Basis qualitativer Abschätzungen Entscheidungen treffen. Eine quantitative Analyse der Prozesskette und der damit verbundenen Daten wird, wenn überhaupt, in der Vorplanung einer Erntekampagne durchgeführt. Hier kommen oftmals digitale Dispositionssysteme ohne spezielle landwirtschaftliche Ausprägung zum Einsatz. Systeme die den Prozessablauf auch zur Laufzeit analysieren und bei Bedarf auf quantitativer Datenbasis neu planen und steuern existieren derzeit nicht. Wohl aber stellen Flottenübersichtsapplikationen mittlerweile planungsrelevante Daten auf allen Maschinen zur Verfügung. Diese auszuwerten ist derzeit Aufgabe der Maschinenbediener. An dieser Stelle setzt das Projekt SOFiA an, in dem es die Maschinen zu sogenannten Smart Objects (SO) erweitert, die auf Basis geteilter Informationen eigenständig Entscheidungen über den Prozess treffen können und diese als Handlungsempfehlungen an den Bediener weiterleiten, welche diese dann ausführen.

4.2.2 Dezentrale Steuerung und Planung

Die Planung logistischer Wertschöpfungsketten (engl. supply chains) wird zurzeit meist über zentrale Steuerungsansätze realisiert. Diese Ansätze bauen auf einen zentralen Knotenpunkt, an den alle Prozessteilnehmer (die ausführenden Maschinen) in einem stern-förmigen Netzaufbau verbunden sind. Der zentrale Knoten übernimmt die Aufgabe die für die Planung relevanten Zielgrößen und Randbedingungen der Prozesskette in einem quantitativen Modell abzubilden [224] und auf diesem zu planen. Zur Vorplanung erweist sich dies als sinnvoll, da bei vollständig vorliegenden Informationen über den Prozess auch mathematisch optimale Lösungen gefunden werden können. Zur Steuerung von verteilten Prozessen, wie in der Landwirtschaft, ist die Verwendung eines zentralen Steuerungssystems von Nachteil: Zuerst müssen alle Prozessinformationen von den Teilnehmern in das zentrale System gesandt werden, und ebenso alle Steuerbefehlen nach der Planung an diese zurück gegeben werden. Insbesondere bei räumlicher Trennung der Prozessteilnehmer in Kombination mit infrastrukturellen Problemen, wie mangelnder Mobilfunkabdeckung, kostet die zentrale Planung mehr Zeit als eine adäquate, zeitnahe Reaktion erlaubt (vgl. Abbildung 4.3, links). Die dynamische Anpassung verteilter Prozesse zur Laufzeit ist mit einem zentralen Ansatz deshalb selten möglich.

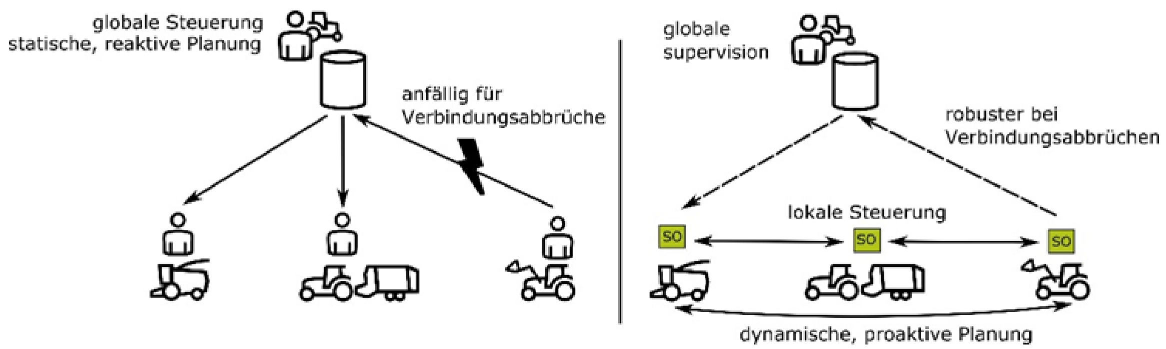


Figure 4.3: Comparison of a central (left) and a decentralized control architecture (right).

Dezentrale Ansätze hingegen beruhen auf der Idee, die Entscheidungsfindung auf die einzelnen Prozessteilnehmer zu verteilen und in räumlich eingegrenzten Teilprozessen durch direkte End-zu-End Kommunikation genügend Informationen zusammen zu tragen, um lokale Prozessoptimierung zu betreiben. Entscheidungen sollen, sofern möglich, unmittelbar dort getroffen werden, wo ein Problem auftritt. Damit eine dezentrale Prozesssteuerung gelingt, ist es also sinnvoll den betrachteten Prozess in möglichst unabhängige Teilschritte zu segmentieren. Bei der Silomaisernte ist dies möglich, da die ortsgebundenen Unterprozesse der Ernte im Feld teilweise unabhängig von denen der Einlagerung am Silo sind und somit auch größtenteils lokal koordiniert werden können (vgl. Abbildung 4.3, rechts). Es wird angenommen, dass jene Prozesse, in denen mehrere Maschinen in räumlicher Nähe interagieren, auch diejenigen sind, die schnelle Reaktionen erfordern und auch auf den Austausch größere Mengen von Daten angewiesen sind. Die Koordination des Überladevorgangs vom Feldhäcksler auf ein Transportfahrzeug, zum Beispiel, erfordert eine zeitnahe, durchgehende Kommunikation zwischen beiden Maschinen zur Koordination von Fahrtrouten, der Steuerung der Überladeautomatik, sowie dem Austausch der übergeladenen Erntemenge. Diese Daten können vollständig lokal, also über direkte Kommunikation mit hoher Konnektivität versandt werden, um die relevanten Informationen für eine lokale Planung und Steuerung auf einer Maschine vor Ort zu aggregieren. Der Umweg über eine zentrale Steuerungseinheit ist hier nicht nötig. Einige Steuerungsvorgänge, wie zum Beispiel das Scheduling der Transportfahrzeuge, müssen jedoch auch über räumliche Distanz hinweg und in einer globalen Sicht auf den Prozess realisiert werden. Auch dies lässt sich in einer dezentralen Architektur lösen, indem einzelne Maschinen als Koordinatoren für eine Reihe von Unterprozessen bestimmt werden, welche die Statusinformationen dieser Prozesse zusammenfassen und gebündelt koordinieren. Die Zuweisung der entsprechenden Rolle ist hierbei nicht statisch festgelegt, sondern wird nach Kriterien wie verfügbarer Konnektivität vergeben und adaptiert. In der Silomaisernte bietet es sich an, jeweils einen Feldhäcksler für alle Prozesse im Feld und ein Verdichtungsfahrzeug für die Prozesse am Silo einzusetzen. Diese aggregieren dann lokal alle relevanten Informationen und prozessieren diese soweit, dass sich Scheduling der Transportfahrzeug, sowie die Koordination von ein- und ausgehendem Massestrom im Ernteprozess auf den Austausch kompakter Kenngrößen, wie z.B. der geplanten und tatsächlichen Ankunftszeiten der Transportfahrzeuge, bzw. Ernte- und Verdichtungsleistung reduzieren lässt. Eine optimale Lösung im mathematischen Sinne lässt sich mit dezentralen Ansätzen zwar nicht immer finden, aber im Hinblick auf die praktischen Anforderungen der Silomaisernte verspricht ein dezentraler Ansatz mit lokaler Optimierung des Prozessablaufs eine wesentlich robustere Steuerung, da der Kommunikationsaufwand und die Latenzen eines zentralen Ansatzes wegfallen.

4.2.3 Projekt SOFiA

Das vom BMBF-geförderte Projekt *Prozessinnovation in Planung und Steuerung von Wertschöpfungsnetzwerken durch Integration von Smart Objects und Smart Finance Ansätzen (SOFiA)*, beschäftigt sich mit der digitalisierten Abwicklung und Abrechnung von logistischen Wertschöpfungsketten. Im Projekt arbeiten das Fraunhofer-Institut für Materialfluss und Logistik, der Landtechnikhersteller CLAAS, der Technologiekonzern Diebold Nixdorf, sowie das Logistikunternehmen EKOL zusammen.

Projektziel ist es ein dezentrales Netzwerk aus Smart Objects (SO) zu erzeugen, das Maschinen, Sensoren und Menschen dezentral vernetzt und ermöglicht, logistische Netzwerke auf quantitativer Basis unmittelbar auf der Prozessebene zu steuern. Hierzu werden die beteiligten Maschinen und z.T. auch Güter (Container), dahingehend digitalisiert und vernetzt, dass sie ihre Rolle im Prozess verstehen und dementsprechend handeln. Hierzu ist es nötig, dass jedes Smart Object in der Lage ist seinen eigenen Prozessstatus zu ermitteln, ihn mit anderen Prozessteilnehmern auszutauschen und in gegenseitiger Abstimmung angemessene Verhaltensweisen zu bestimmen. Diese werden anschließend als Handlungsempfehlungen den ausführenden Maschinenbedienern oder Disponenten präsentiert. Solche Empfehlungen können im kurzfristigen Fall das anpassen einer Transportgeschwindigkeit oder des Produktionsdurchsatzes sein, mittelfristig aber auch Anpassungsvorschläge für die Kettenkonfiguration sein (z.B. Hinzunahme einer weiteren Transporteinheit). Ziel ist es Logistikprozesse reibungsloser und somit effizienter zu steuern, und somit unerwünschte Effekte wie eine mangelhafte Auslastung von Kapazitäten, vermeidbare Stückkosten und mangelnde Termintreue zu reduzieren. Neben der Steuerung des logistischen Ablaufs soll ebenfalls eine vollständige Erfassung von Leistungsdaten den anschließenden Administrationsvorgängen erleichtern.

Hardwareseitig wird zur Umsetzung der Smart Objects im Projekt auf kleine eingebettete Systeme (Raspberry Pi) gesetzt, die aufgrund des steigenden Leistungszuwachs und Preisverfall eine geeignete Forschungsplattform bieten. Softwareseitig folgt die Umsetzung der Smart Objects dem Agentenansatz (vgl. [92]) und der Idee, dass sich die Teilprozesse einer logistischen Kette als Aufträge beschreiben lassen und deren Ausführung von den Smart Objects untereinander verhandelt wird. Hierzu werden die Aufgaben mit Zielvorgaben (z.B. verfügbare Zeit, erlaubte Kosten, angestrebte Qualität) versehen und hierarchisch organisiert, sodass die Erfüllung der Ziele von oben nach unten delegiert wird und die Informationen über die Ausführung von unten nach oben aggregiert werden. Um sowohl den Prozess, als auch die Maschinen und deren Umgebung zu modellieren, werden ontologische Wissensrepräsentationen auf Basis von Semantic Web Technologien (RDF, OWL) eingesetzt. Diese Modelle werden anschließend über regel-basierte Inferenzmechanismen (SWRL) einem Soll/Ist-Vergleich von Prozesszuständen und Zielvorgaben unterzogen und bei Steuerungsbedarfen mit einem ereignis-diskreten Simulator für logistische Netzwerke verbunden [140]. Dieser simuliert dann verschiedene logistische Szenarien, welche anschließend ausgewertet werden und als Grundlage für Handlungsempfehlungen dienen. Zur Verbindung der Smart Objects werden hybrid-opportunistische Netzwerke und verzögerungstolerante Kommunikation basierend auf dem Store-Carry-Forward Prinzip genutzt.

Die hier präsentierten Ansätze einer dezentralen Prozesssteuerung werden im weiteren Fortschreiten des Projekts SOFiA erprobt, erweitert und daraufhin bewertet, ob die erhofften Verbesserungspotenziale für die Planung und Steuerung logistischer Prozesse in der Maisernte erzielt werden können.

Chapter 5

Grounding Semantic Maps in Spatial Databases

Henning Deeken Knowledge-Based Systems Group, Osnabrück University, Osnabrück, Germany
Thomas Wiemann Knowledge-Based Systems Group, Osnabrück University, Osnabrück, Germany
Joachim Hertzberg Knowledge-Based Systems Group, Osnabrück University, Osnabrück, Germany
DFKI Robotics Innovation Center, Osnabrück Branch, Osnabrück, Germany

Abstract

Semantic maps add to classic robot maps spatially grounded object instances anchored in a suitable way for knowledge representation and reasoning. They enable a robot to solve reasoning problems of geometrical, topological, ontological and logical nature in addition to localization and path planning. Recent literature on semantic mapping lacks effective and efficient approaches for grounding qualitative spatial relations through analysis of the quantitative geometric data of the mapped entities. Yet, such qualitative relations are essential to perform spatial and ontological reasoning about objects in the robot's surroundings.

This article contributes a framework for semantic map representation, called SEMAP, to overcome this missing aspect. It is able to manage full 3D maps with geometric object models and the corresponding semantic annotations as well as their relative spatial relations. For that, spatial database technology is used to solve the representational and querying problems efficiently. This article describes the extensions necessary to make a spatial database suitable for robotic applications. Especially, we add 3D spatial operators and a tree of transformations to represent relative position information. We evaluate the implemented capabilities and present real life use cases of SEMAP in different application domains.

5.1 Introduction

A semantic map for a mobile robot has to combine semantic, topological and geometric information in a compact representation. These different types of information are required to solve relevant problems like localization, path planning, 3D trajectory planning, task execution, object search, and more. Hence, semantic maps have to evolve from specially tailored task-specific representations towards multi-purpose environment models that can be re-used in different applications and updated dynamically. Such generalized models should be able to fuse information from different data layers via a query interface that allows to extract task-specific environment data on-demand.

Current approaches in semantic mapping already exhibit features of more generalized environment models. There has been significant progress in describing the semantics of environments using ontological approaches to model a-priori background knowledge and to capture facts about an environment's current state. Similarly, large-scale spatial mapping, scene segmentation, and object recognition are well understood and can be used to gather spatio-semantic data of real-world environments. The study of the anchoring problem [70] has lead to effective strategies to derive environment knowledge from sensor data and to track entities and their features over time. To that end, it is crucial to link semantic knowledge with geometric data and perform data analysis across both domains dynamically with the acquisition of updated information. However, the representational frameworks underlying semantic maps are still unable to ground spatial relations between entities. If grounding spatial relations is addressed, it is usually done during semantic map building. Appropriate tools on a representational level are rarely seen, although the benefit of spatial analysis for enriching semantic knowledge – especially for anchoring physical objects in large-scale semantic maps – is obvious.

This article presents how to derive and manage qualitative spatial relations between objects from quantitative geometric environment data captured by some kind of mapping approach. It shows how to realize efficient spatio-semantic querying on semantic maps by integrating a spatial database into a semantic mapping framework. The close integration of a spatial database provides a dedicated storage and processing module for the spatial environment data as a suitable complement to a classical knowledge-based system. By correctly anchoring spatial records to their respective semantic counterparts, the database's spatial operators provide the ability to derive qualitative information about the spatial relations between stored entities that is otherwise covert. This adds an essential feature to semantic map representations, since grounding spatial relations uncovers important information about the robot's environment. In our approach, the current semantic world model stored in a dedicated knowledge base can be updated accordingly whenever an object is inserted or modified in the semantic map. It also allows to query for environment data on demand using spatial and semantic constraints simultaneously, which allows to answer typical questions about the environment, as presented in Figure 5.1.

We have cast this approach to combine semantic and spatial data into the *Semantic Environment Mapping Framework (SEMAP)*. In this paper, we describe the basic concepts of SEMAP's architecture, with special focus on the integration of the spatial database into the semantic mapping framework. We discuss the extensions added to an existing geometric database system that are necessary to achieve the desired functionality. We present and evaluate the new features of this semantic mapping framework that arise from the novel combination of the geometric database with a classical knowledge-based system, especially the feature of grounding qualitative spatial relations through the quantitative analysis of spatial data. We show that the presented approach generalizes well into different application domains by presenting real world examples of applying SEMAP.

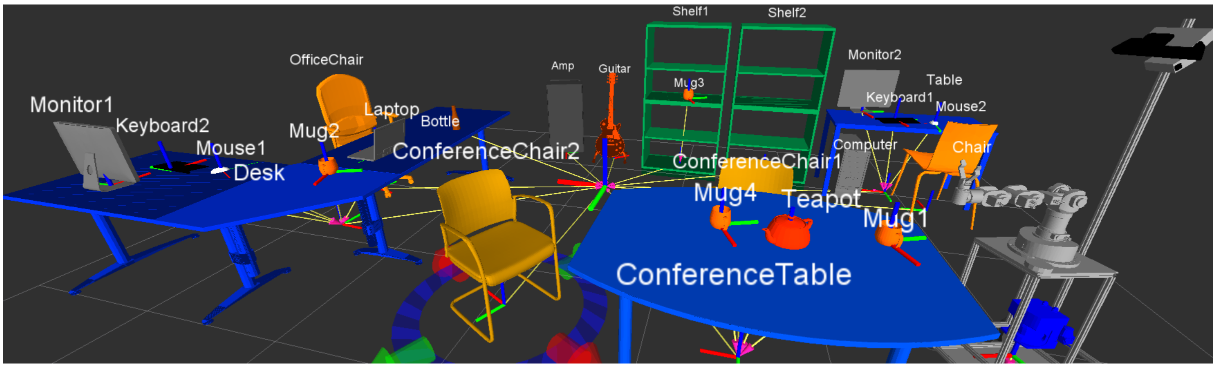


Figure 5.1: Artificial rendering of an office environment modeled in SEMAP. The labels denote object instances that are present in the knowledge base at their locations in 3D space and in relation to other objects, as represented in the semantic map.

- **Q1** Which objects are in this `Room`?
- **Q2** How many `Chairs` exist?
- **Q3** Where is `Mug2`?
- **Q4** Which `Mug` is closest to the robot?
- **Q5** Is there a `Computer` in this `Room`?
- **Q6** Is the `Monitor1` on the `Desk`?

5.2 Related Work

Over the last decade, the discipline of semantic mapping has become increasingly popular and successful. A recent survey by Kostavelis and Gasteratos [128] reviewed more than 120 different approaches. It summarizes the significant progress made on a broad range of mapping approaches and applications for semantic maps, including task planning [100], localization [170, 232], navigation [45, 71] and human-robot-interaction [21].

This review also revealed a significant heterogeneity in the processes of semantic map building, as well as in the underlying semantic map representations, because access to spatio-semantic environment data is beneficial in a multitude of applications. But the level of detail or selection of appropriate data types and information sources varies significantly, depending on the application. Therefore different semantic maps use different underlying spatial representations and semantic annotations.

Bastianelli et al., for example, presented a hybrid semantic map consisting of annotated 2D occupancy grids, whose labels were given by a human instructor, and topological graphs [21]. It was used for topological navigation, object search and object manipulation. Nüchter and Hertzberg demonstrated how 3D point clouds can be automatically segmented into categories like walls, floor and ceiling [152]. Pronobis and Jensfelt presented a vision-based system that allows to identify objects and rooms by analyzing features on position-tagged images and the geometric attributes, like area and shape, of occupancy grid maps [171].

This heterogeneity is also reflected in the definition of semantics maps, which either intentionally make no particular assumption about the mapping process or the underlying representations [136, 152] or rely on the concept of hybrid maps [56, 99]. Yet, a common agreement is that semantic maps have to be paired with formal knowledge representations and reasoning, to unfold their full potential. Recent literature provides several examples of how knowledge base components can be beneficial in semantic mapping [49, 57], for reasoning about the environment. These approaches usually use ontological and graph-based knowledge representations, based on description logics [16].

One example for such a system is *KnowRob* [208], which combines a knowledge representation in the Web Ontology Language (OWL) and Prolog-based reasoning with an interface to the robot's control architecture. The goal of KnowRob is to provide a system that is fully integrated with the robot to generate new knowledge from sensor perception and effectively guide the robot's behavior through semantic inference. In the context of semantic mapping it has been used to answer queries about a semantic object map [160]. It has been used in various projects. One is *RoboHow* that explored possibilities to use the World Wide Web as resource to find instructions for solving everyday manipulation tasks [221]. Another example is *RoboSherlock*, which defines a generic interface for perception algorithms and a knowledge base to plan which perception modules to use and to consistently feed perceptions into the knowledge base [26]. *OpenEASE* aims at creating a knowledge base for manipulation episodes that can be queried by multiple robots to share their experiences in order to learn manipulation tasks and to improve their performance [25].

It is apparent that semantic maps are intertwined with knowledge representations and reasoning capabilities. It is, however, somewhat unclear where to draw the line between a semantic map and the associated knowledge representation and reasoning systems. Similarly, managing their inter-dependencies remains an open issue. In [128] Kostavelis and Gasteratos regarded the question of *How semantic maps aid knowledge representation and vice versa?*, as one of the open topics in semantic mapping. They pointed out that ontologies and other formal knowledge representation schemata can yield additional insights into a model of the robot's surroundings by encoding and revealing attributes even when these are not perceivable. However, the authors stressed that proper semantic mapping fuels the knowledge representation by recognizing and anchoring entities in the environment to connect spatial and semantic knowledge. For that, they considered creating a *spatially ordered hierarchy* important. This assessment directly points to the challenge of continuously grounding the spatial relations of objects within an environment.

The set of qualitative spatial relations holding in the environment's current state, such as "`Mug2` rests **on** `Desk`" or "`ConferenceTable` is **in front of** the `Robot`", has to be uncovered by inspecting the environment's spatial aspects. To logically reason about the spatial relations between entities by using qualitative spatial reasoning (QSR), they need to be explicitly stored as symbolic knowledge. Qualitative constraint calculi, like the interval calculus [11] or the Region Connection Calculus (RCC) [68], can effectively reason about sets of qualitative spatial relations. Suitable software solutions like the SparQ toolbox [228] exist, but are rarely integrated into semantic mapping approaches. According to Wolter and Wallgrün, this is due to a lack of explicitly available qualitative spatial relations, since the important step called qualification is often missing and remains largely unsolved in practice. The lack of effective tools for grounding spatial relations in sensor data captured from the real physical environment inhibits a wide-spread use of QSR in robotics.

Uncovering spatial relations can be part of the map building and anchoring process. Sjöö et al. presented a combination of an axiomatic system and probabilistic inference to interpret topological spatial relations such is-on or is-in during the mapping process [190]. For additional examples of reasoning with spatial relations in the context of real-world robotics applications, we refer to the comprehensive review by Landsiedel et al. [135].

Grounding spatial relations during the map building pipeline is generally a good approach, but is restricted to processing incoming sensor data and limited to the current excerpt of the environment that is under the robot's scrutiny. Hence, it usually does not scale over the entire environment model, nor does it allow to make spatial queries for objects, whose spatial relations are not yet grounded. Especially, when environment dynamics are considered and a large volume of spatial and semantic data has to be integrated into the semantic map on a continuous basis, maintaining a set of geometrically grounded spatial relations in the knowledge base becomes a tedious task. Hence, effective tools to map from quantitative metric data to qualitative symbolic facts are necessary in the context of semantic mapping, in order to enable the usage of qualitative spatial calculi or other types of formal reasoning over spatio-semantic environment data. It is therefore desirable to provide the capability of grounding spatial relations as a feature of the semantic map, since this complements the handling of spatial relations during map building. In this article, we propose to use a spatial database as a tool to map from geometric data to symbolic spatial relations.

Spatial databases extend relational databases to store, query and analyze geometric data. They enable spatial lookup to search for geometries within a certain region or volume and provide spatial analysis to test if two geometries overlap or intersect. To reduce the evaluation time of spatial relations, spatial indexing techniques are used. Spatial indexing abstracts complex geometries to primitive bounding geometries (2D rectangles or 3D boxes), whose relations can be evaluated efficiently even in large data sets. Most indexing techniques rely on height-balanced search trees of bounding geometries, so called R-trees [109].

Spatial operators determine whether a spatial relation holds between two geometries and map from quantitative geometric data to symbolic spatial predicates. Evaluating distances in 2D and 3D is straightforward, but the analysis of topological and directional relations is subject to extensive research, especially in 3D. Topological relations in 2D have been extensively studied. The DE-9IM model [67, 89] is the standard for spatial databases proposed by the Open Geospatial Consortium (OGC) [69]. An overview of approaches to address 3D topological analysis is given in [235] based on the geometric decomposition scheme presented in [47, 74], to realize the evaluation of 3D intersection, touch and containment. The research on qualitative spatial reasoning (QSR) has proposed various calculi to define and work with directional relations, varying frames of reference and cardinal directions. A comprehensive overview of one and two dimensional solutions is given in [174]. For 3D, Borrmann and Rank describe two approximate approaches using *projection-based* and *half-space* models to analyze directional relations [48].

Spatial databases are commonly used as back-ends for geographic information systems (GIS) in geography, climatology and governmental administration, to store and analyze geographic and cartographic data. GISs primarily offer processing for 2D data, but 3D is actively studied ([6, 50]) and modern solutions provide at least storage for 3D data. However, a full tool set of spatial operators in 3D is still missing.

Since spatial databases already integrate means for spatial analysis on top of storing geometric representations, they are apt candidates for determining qualitative spatial relations in the context of semantic mapping. Therefore, the main contribution of this article is to solve the open problem of grounding qualitative spatial relations in semantic maps by integrating a spatial database into a semantic mapping framework.

We analyzed existing spatial databases and identified the extensions that are needed to make spatial relations qualitative for 3D objects. Besides extending a spatial database with new operators, we present the corresponding schemas and table layouts that are required to support articulated objects and dynamic update of spatial relations when objects are inserted or deleted. Our implementation focuses on making spatial relations qualitative to update the current state of the environment. It serves as a means to generate symbolic knowledge about known facts and spatial relations about the most likely world model. Although probabilistic mapping approaches can be used to determine the current world state modeled in SEMAP, they are not yet considered explicitly in the current implementation.

If the robot's perception provides information on changes in the environment, SEMAP's model can account for these dynamics by adding, deleting or updating its entities. The framework currently does not account for a history of the environment's past states, nor does it provide a set of alternative environment models or a probability distribution over models, to account for uncertainties during the map building process. From a probabilistic perspective, SEMAP represents a maximum likelihood model that is maintained over time.

We illustrate the steps necessary for this integration, based on our proof-of-concept implementation and an exemplary office domain. More domains are presented and discussed in the application examples and in the final discussion.

5.3 The SEMAP Framework

SEMAP was designed as a representation and reasoning system for environment modeling in robotics. It is based on an object-based environment model in which every entity in the environment belongs to a known concept class, contributes to a set of asserted facts and consists of a spatial model, which can be either a single volumetric body or an articulated kinematic chain of those. To account for the different nature of symbolic and geometric data, SEMAP stores the different kinds of information in dedicated storages. A close connection between the spatial and semantic aspects of an environment is maintained by the framework's spatio-semantic data maintenance layer and querying interface as shown in Figure 5.2.

5.3.1 Architectural Concept

All geometric aspects are stored in a PostGIS database and describe the shapes and poses of the individual objects in the environment. For articulated objects, their kinematic chains and current joint configurations are represented as well. Additionally, the database maintains relational links that connect geometric data sets to their complementary semantic descriptions in a separate knowledge base with factual and conceptual environment information.

The knowledge base uses description logics (DL) [16], featuring the classical separation into a T-box for storing concept definitions including the taxonomy and an A-box for asserted facts. We use a DL-based approach because the underlying ontological models can be constructed to separate domain-independent and domain-dependent knowledge. This helps make the core components application-independent and extensible to different application domains. For that,

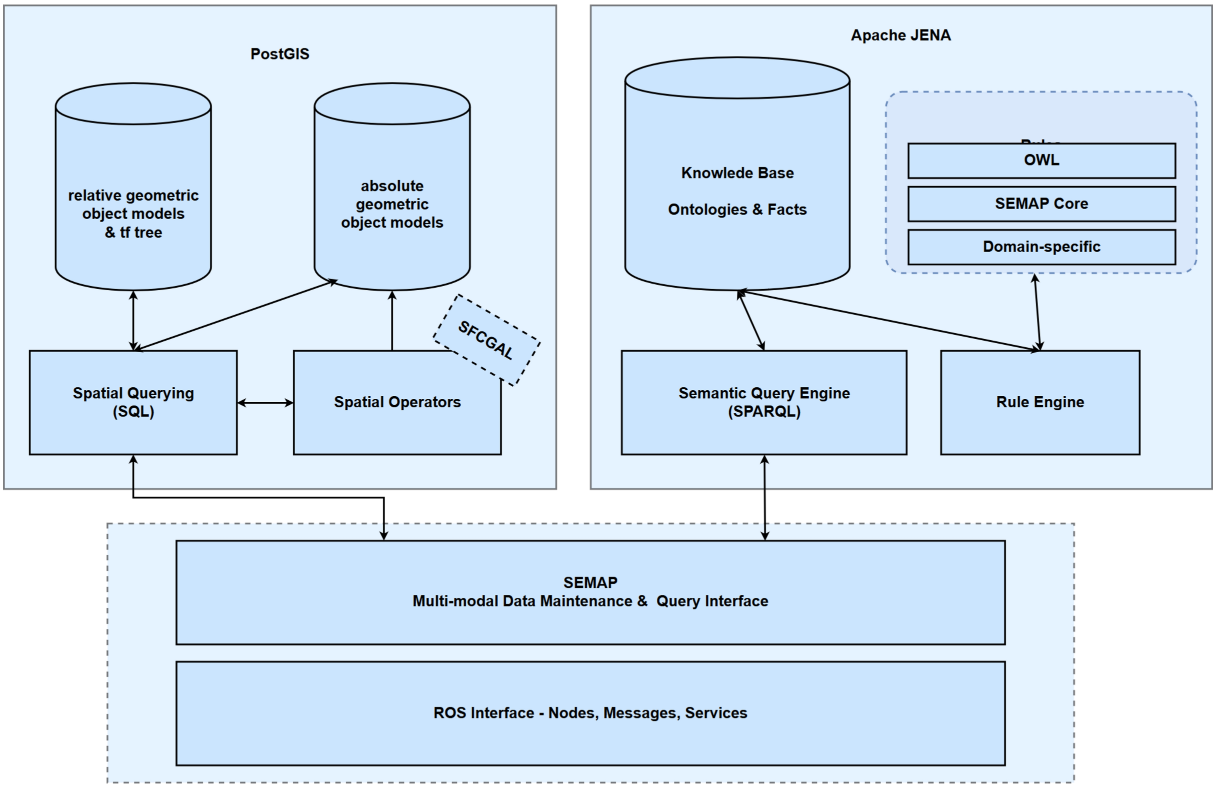


Figure 5.2: The SEMAP framework consists of a PostGIS database which provides spatial data storage and querying capabilities and an Apache JENA triplet store to maintain the ontological background knowledge and actual facts. Both data domains are coupled via a query interface that can be accessed by robot control systems like ROS.

the T-box maintains a set of domain-independent ontologies that provide a semantic model of the supported geometric types, how they can be combined to form objects and how objects constitute an environment, whereas a domain-dependent ontology provides the necessary vocabulary to describe knowledge about a certain application. Within the A-Box the combined ontological descriptions are used to store facts on individual instance in the environment. Such a system can easily be paired with reasoning modules to enable rule-based inference on the stored environment knowledge.

To communicate with robot control frameworks, we use an intermediate layer between the robot's control architecture and the semantic map representation. This layer provides interfaces to insert information about environment entities from different data sources and handles updating the model. It links the spatial database to the knowledge base by adding URIs to the geometric entities stored in the relational data base that point to the respective instances in the knowledge base. This interface layer handles the incoming queries to retrieve target-specific data and convert it into the required representation.

5.3.2 Software Components

To represent geometries we chose to use PostGIS as it supports 3D geometries best among the various open source spatial data base implementations available, as shown in Table 5.1. PostGIS is an open source GIS, based on the relational database PostgreSQL [167], that is compliant with the standards of the Open Geospatial Consortium (OGC). PostGIS provides representations for a number of geometric primitives. These include points, lines, polygons, and collections of geometries, as defined in the "Simple Feature Access" specification [69]. Even though the standard is specified for 2D geometries only, PostGIS also supports three dimensional primitives and includes data types for meshed surface structures based on triangular or polygonal primitives. PostGIS's analytic functions can interpret the spatial information as geographic data in a geodetic reference system or as geometric data in Cartesian space. For spatial querying, PostGIS combines regular R-trees with Generalized Search Tree indices (GiST) to speed up mixed queries with spatial and relational constraints. To analyze 2D simple feature geometries, PostGIS uses the GEOS library [157], which provides an extensive tool set of spatial operators. Native PostGIS only supports few operations on 3D data, but can be extended with custom operators using the SFCGAL plugin [158]. The SFCGAL project defines an interface to the Computational Geometry Algorithms Library (CGAL) [212], which provides an extensive set of geometric algorithms. These algorithms can then be used to define additional 3D spatial operators for PostGIS. PostGIS in combination with the SFCGAL extension realizes the storage of spatial environment data consisting of both 2D and 3D geometric primitives. For spatial analysis, the close integration of CGAL allows the missing spatial operators to be implemented for 3D geometries.

In addition to using PostGIS, we have implemented a prototypical integration of the semantic web framework Apache JENA to support query languages like SPARQL. We will not dive into the details of this approach – as it is work in progress – but present a preliminary example of the ontology that will be used to link the spatial database to the knowledge base.

To demonstrate the use of SEMAP on a real robot, we implemented an interface to the Robot Operating System (ROS). This will be made public as an addition to SEMAP together with the ROS bindings and the reference data set presented in this article.

5.3.3 Ontological Model

The ontological model underlying a environment representation in SEMAP is comprised of two parts: SEMAP's core ontology, which is independent of any domain specific application and a domain-specific ontology, which may be changed depending on the application.

SEMAP's core ontology gives the conceptual background for representing the spatial elements within an environment model as presented in Figure 5.3. These concepts are closely related to the data base layout of the PostGIS back end, as will be discussed below. The ontology uses standards from the Open Geospatial Consortium (OGC), because these well-defined models of geo-spatial data are in alignment with PostGIS's data types, which were also defined by the OGC. GeoSPARQL's `SpatialObject` and the fundamental distinction between geometries and features are integrated in SEMAP's upper ontology. Here, the concept `Geometry` describes any kind of spatial primitive and provides a semantic wrapper for all OGC data types and serves as

Name	**PostGIS**
Description	spatial database extension for the PostgreSQL database
Authority Website	`http://postgis.net`
Spatial Types	Points, LineStrings, Polygons, MultiPoints, MultiLineStrings, MultiPolygons, GeometryCollections, Triangle Irregular Networks, Polyhedreal Surface
Spatial Index	R-tree-over-GiST spatial indexing for high-speed spatial querying
Spatial Functions	Over 300 functions and operators, no geodetic support except for point-2-point non-indexed distance functions, custom PostGIs for 2D and some 3D, some MM support of circular strings and compound curves
Name	**MySQL**
Description	Includes a limited set of spatial representations and queries natively.
Authority Website	`http://www.mysql.com`
Spatial Types	Geometry, Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon, GeometryCollection
Spatial Index	R-Tree quadratic splitting-indexes only exist for MyISAM
Spatial Functions	OGC mostly only MBR (bounding box functions) few true spatial relation functions, 2D only
Name	**Spatial Lite**
Description	SQLite with spatial datatypes, functions, and utilities
Authority Website	`https://www.gaia-gis.it/fossil/libspatialite/home`
Spatial Types	Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon
Spatial Index	R-Tree variants
Spatial Functions	Basic functions for Point, LineString and Polygon

Table 5.1: Comparison of open source spatial database implementations regarding their spatial types, spatial indexing technique, and available spatial operators. Adapted from `http://infolab.usc.edu/csci587/Fall2016/`

a bridge to the well known Simple Feature Ontology. SEMAP's KB contains a corresponding instance of a `Geometry` sub-concept, for every geometric primitives stored in SEMAP's DB. The property `semap:hasDbId` is used to create an associative link between the geometric primitive and its semantic wrapper. SEMAP internally uses these associations to join spatial and semantic data, in its query interface.

The super-concept `Feature` is used for all things that can be described spatially like SEMAP's `ObjectModel`, which aggregates sets of semantically wrapped geometries to represent an object. For this, it uses the `geo:hasGeometry` property and its two specializations: `semap:hasBody` composes a set of geometries that constitute the object's actual body. In case of articulated objects, the `Link` and `Joint` concepts are used to describe the object's kinematics. `semap:hasAbstraction` provides a set of coarser representations, like oriented and axis-aligned bounding boxes and convex hulls. These abstractions are used for accelerated spatial processing and enable the analysis of directional relations like `left-of` or `above-of`, based on projection and halfspace geometries [46].

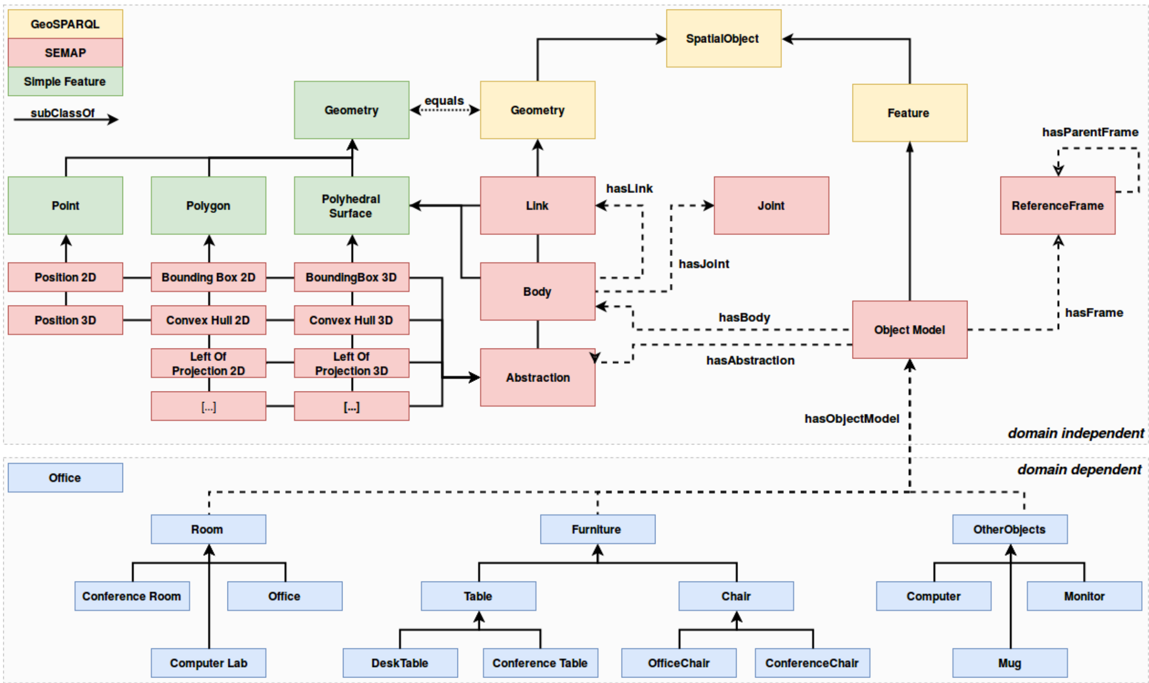


Figure 5.3: An excerpt from the ontology that is used to fuse the labels of objects stored in the spatial data base with semantic concepts in different domains.

To create a spatio-semantic environment model for a particular application, domain-specific ontologies, knowledge bases and rule-sets can be imported into SEMAP's knowledge base component. To describe domain-specific concepts spatially and reason about them as part of SEMAP's environment model, the respective entities can be associated with an `ObjectModel` via the `hasObjectModel` relation. Figure 5.3 shows this by connecting objects from a simple ontology describing objects and rooms in an office environment to the SEMAP core ontology. The used ontology is in partial alignment with the indoor furniture classification ontology used in our previous works on semantic mapping [108].

5.3.4 Database Schema

Figure 5.4 displays the database schema for storing semantically annotated objects in the spatial database. This schema is roughly divided into three parts: the representation of object *classes* (red), individual object *instances* (blue), and their different *geometric* representations at different abstraction levels (green).

To connect the geometric models in the database and the conceptual representation in the ontology, the entity ObjectDescription has an attribute ObjectClass that maps the description in the database to one of the concepts in the ontology (ie. to the concept `office:Mug`. To represent articulation, each object class can consist of several Links and Joints that are connected in a kinematic chain. The individual ObjectInstances have individual Names to have a readable label besides the internal ID, which is aligned with the semantic wrapper's

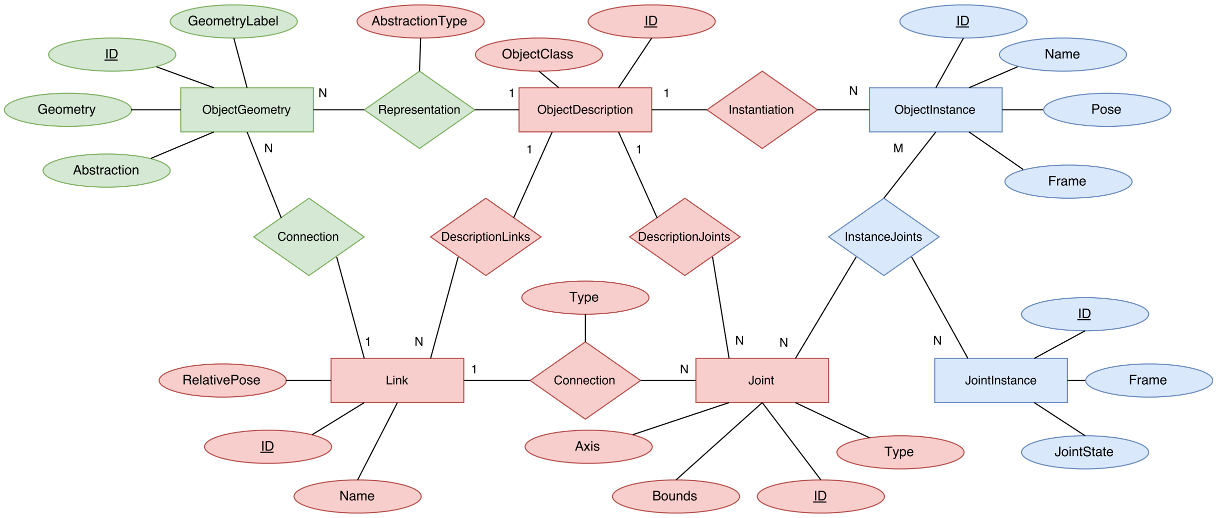


Figure 5.4: The database schema for representing the environment model. Explanations provided in the text.

`hasDbID` property. To model articulation, each object instance can represent individual JointInstances that are linked to connecting ObjectGeometries via JointConnections and LinkConnections that refer to the object descriptions links and joints.

In our modeling, the ObjectDescription entity represents the generic spatial model of an object class that can be instantiated via the Instantiation relation. Since the individual attributes are stored in the blue instantiation relations, the geometries associated with the object descriptions can be re-used to prevent storing identical geometries multiple times. SEMAP supports 3D polyhedral mesh data to describe the body geometries of each individual part of an object. The individual configurations of the partial geometries are transformed according to the instances' poses and joint states. Since geometric queries in 3D can be computationally expensive, we can store object geometries at different *abstraction* levels. For example, the precise polyhedral mesh representation of a CAD model can be abstracted by its bounding box or convex hull, which can be used for efficient but less precise qualification. These abstractions are initialized when the objects are inserted into the database and updated dynamically. Examples of the computed abstractions are shown in Figure 5.5. SEMAP's default abstractions are 2D and 3D axis-aligned and oriented bounding boxes, and convex hulls. Additionally, point-based abstractions are also computed. These auxiliary geometries are created with functions from PostGIS and SFCGAL for the entire object as well as for each individual link. The level of abstraction is stored in the attribute AbstractionType in the Representation relation. By convention, all object geometries are defined in a right-handed coordinate system and the base link of an object is placed at the object's bottom, as it is often done when using the Unified Robot Description Format (URDF). For convenience, SEMAP supports the direct import of URDF files.

Semantic information about a geometry is stored in a GeometryLabel string that labels the sub-part of the object. These refer to a semantic description that is maintained separately in the dedicated knowledge database and linked to the spatial database table via this label. This way, the semantic description of the object is directly integrated into the PostGIS database,

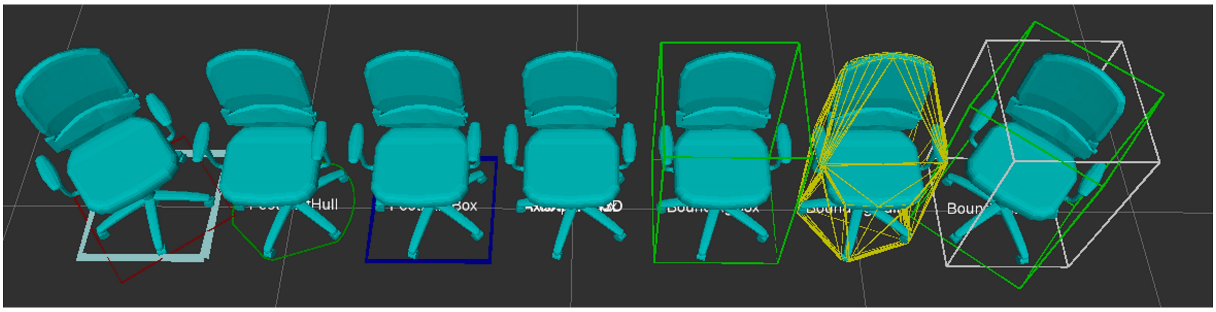


Figure 5.5: SEMAP provides a set of geometric abstractions to enable accelerated spatial queries. From left to right: 2D axis-aligned bounding box, 2D convex hull, 2D bounding box, 3D bounding box, 3D convex hull, and 3D axis-aligned bounding box. The axis-aligned bounding boxes (in gray) are overlaid with the oriented bounding boxes for comparison.

so that we can use relational queries on these labels to emulate data retrieval based on object semantics. We use this feature to perform spatial queries in PostGIS to ground certain spatial predicates, which are then asserted to the knowledge base as facts.

As described so far, the object descriptions are only the blueprints from which instances are created to model the actual environment. To build an actual environment model, the ObjectInstance table combines a reference to an object description with the position and joint states of an actual instance. To manage positional information within the environment model, SEMAP implements a relative positioning system using a transformation tree. Frames in the transformation tree span local coordinate systems, in which the relative positional information is expressed. These frames are defined with respect to each other and create a directed tree. At the root of the tree, the global root frame defines the global coordinate system. The relative frames of all objects can be transformed into this system by traversing the tree. With this transformation tree, SEMAP also supports a common practice in many robotic systems (in analogy to ROS's tf library), but in persistent storage. That allows to preserve the environment's state during robot downtime, which is required in long-term applications.

The implementation of the transformation tree is realized in the Frame table, which Figure 5.4 does not show for sake of readability. This relation connects the Pose of an object part's instance to the frame to which it is related, via a reference to the frame of the parent object. Each object instance has a pose, which is the anchor for the object's base link. Additionally, each joint instance has a frame to allow for a frame-based view on the object's entire kinematic chain. Another important function of SEMAP's transformation tree is to build a bridge between two different views of an instance's spatial representation.

Up to this point, we have described the relative view, which is taken in the context of a frame-based positioning system. However, once an object instance is subject to SEMAP's spatial query system, there is also the demand for an absolute view on the object's geometry, because relative geometric information can not be processed by PostGIS's R-tree implementation. In PostGIS, all geometries have to stem from the same global reference frame. In order to obtain reasonable results in the spatial analysis, SEMAP maintains a second object description for each object instance that provides a copy of the relative description's geometries and abstractions in absolute coordinates. To create this view, the transformation from the root frame to the instance's frame is applied to all the geometries stored in the relative description. Since this is a potentially expensive operation, SEMAP creates full absolute representations only on demand. By default, only the description's abstractions are transformed. All absolute representations are cached and reused, until they expire, which happens every time the object changes in pose or configuration. Since an instance's frame can be the reference frame for other objects, any change affects all objects that descend from it.

5.3.5 Spatial Operators

Next, we will review the spatial operators available in PostGIS and discuss their usage in robotic applications. We will distinguish them by their support for the following datatypes: basic 2D and 3D geometric primitives (points, lines, polygons) and 3D triangle and polygonal mesh data. We also discuss how to construct custom operators using the SFCGAL plugin for operators that are missing in native PostGIS, but are required for robotic applications.

Metric Operators

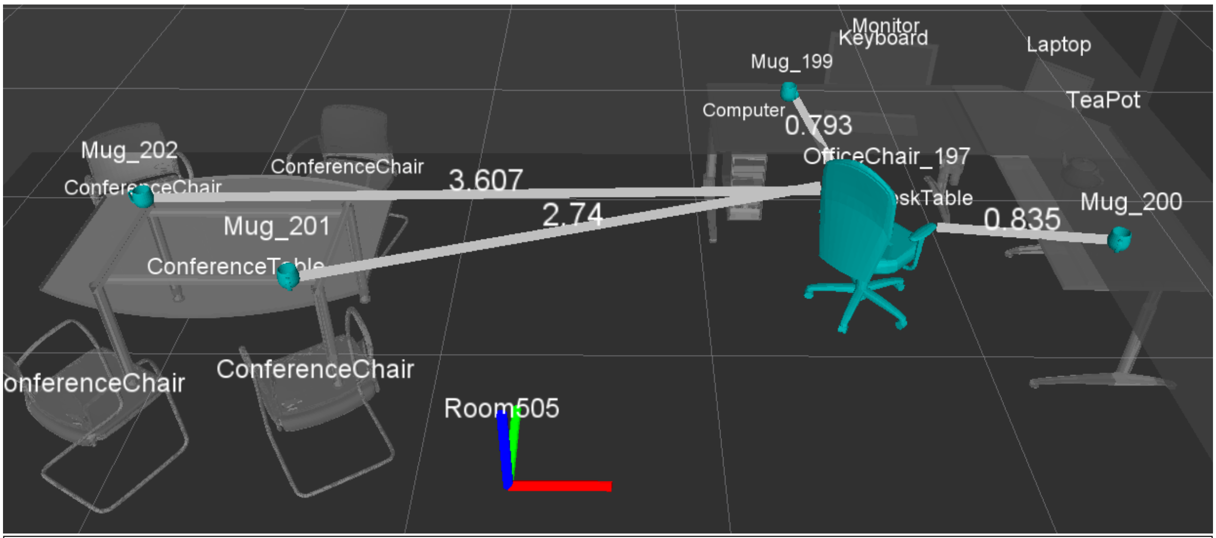
PostGIS offers a number of metrical operators to measure the minimal and maximal distance between geometries, to test whether a geometry is (fully) within a parametrized range of another geometry or not and to return the shortest or longest line between two geometries. These operators are available for most 2D and 3D geometries, except for the TIN type, which is implemented as SFCGAL extension that offers minimal distance measures. A list of all operators available in SEMAP is presented in Table 5.2.

In robotic applications, metric operators are a valuable tool to look up objects within a certain range around a query location, such as the robot's position. SEMAP utilizes the different operators to provide distance-based queries across the various geometric representations of object models. Figure 5.9 shows the implementation of SEMAP's operator to measure the minimal or maximal distance between objects. It allows to sort results in ascending or descending order, which is beneficial in prioritizing objects. Figure 5.6 illustrates the usage of the operators using 3D convex hulls.

Operator	**2D Geometries**	**3D Geometries**	**3D TIN**	**3D Polyhedron**
MinDistance	✓/ ❍	✓/ ❍	❍	✓/ ❍
MaxDistance	✓	✓	✗	✓
WithinRange	✓	✓	✗	✓
FullyWithinRange	✓	✓	✗	✓
ShortestLine	✓	✓	✗	✓

✓: native PostGIS ❍: SFCGAL plugin ✗: currently not implemented

Table 5.2: List of PostGIS's metric spatial operators and the geometric primitives supported.




```
rosservice call /get_distance_between_objects
"reference_object_types: ['OfficeChair'] reference_object_geometry_type: 'BoundingHull'
target_object_types: ['Mug'] target_object_geometry_type: 'BoundingHull'
sort_descending: false max_distance: false"
```

Figure 5.6: Example of using SEMAP's ROS interface to measure the minimal distance between all `OfficeChair`s and all `Mug`s using the 3D convex hulls.

Topological Operators

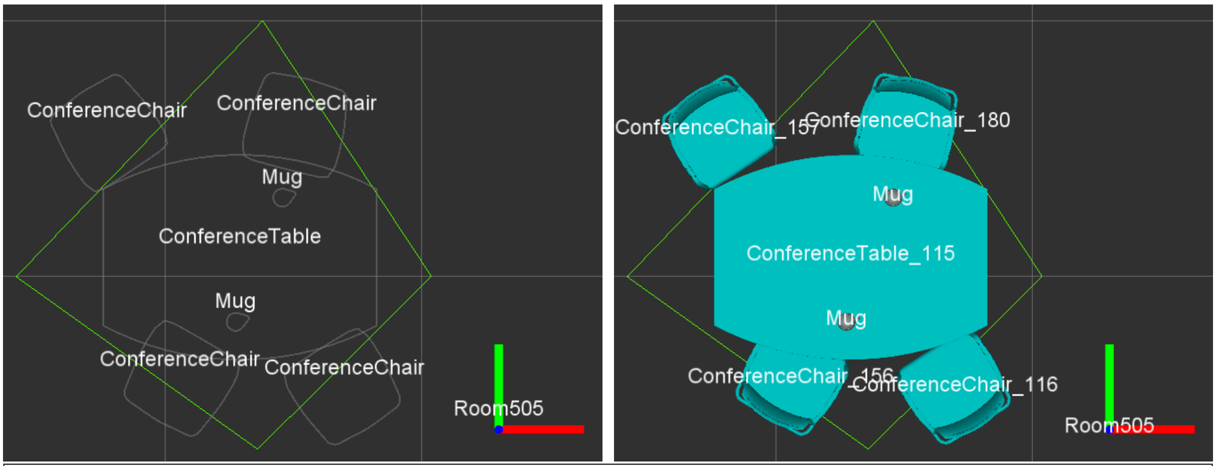
PostGIS implements the DE-9IM calculus [67]. Most common topological relations, such as equals, intersects, covers and touches, can be evaluated for 2D geometries. Similarly, equality, intersection and containment tests for the Box2D type and an intersection test for the Box3D type are available. Additionally, SFCGAL provides intersection tests for all 3D geometries as shown in Table 5.3.

Among these operators, those evaluating containment and intersection relations are most valuable and versatile for robotic applications. On the one hand, they allow for spatial look-up by identifying if an object's geometry lies within (or at least intersects with) another geometry. In this aspect, containment operators work similar to metric operators, but exceed them in flexibility,

Operator	2D Box	2D Geo.	3D Box	3D Geom.	3D TIN	3D Polyh.
Containment	✓	✓	✚	✗	✗	✚
Intersection	✓	✓	✓	❍	❍	❍
Touch	✗	✓	✗	✗	✗	✗
Equality	✓	✓	✗	✗	✗	✗

✓: native PostGIS ❍: via SFCGAL plugin ✚: custom extension ✗: currently not implemented.

Table 5.3: List of available topological spatial operators and the geometric primitives supported.




```
rosservice call /get_objects_within_area
"reference_polygon: ...  target_object_types: ['Chair','Table']
target_object_geometry_type: 'FootprintHull'  fully_within: false"

target_ids: [115,156,116,157,180]
```

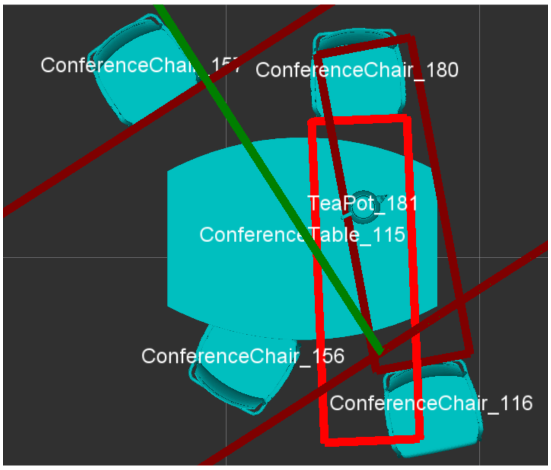
Figure 5.7: The presented operator evaluates the object's 2D convex hulls against a reference polygon. Since the operator's relaxed interpretation was chosen, intersecting objects were included in the query result, as well as those fully within the reference polygon.

since potentially arbitrary areas or volumes can be queried. On the other hand, they allow to ground the spatial predicates that hold between objects, which makes topological operators highly relevant for applications in semantic mapping. By applying topological operators on SEMAP's environment model, all objects that are `in` a certain area can be queried to create the respective semantic knowledge, which in turn can then be processed by qualitative spatial reasoning techniques separately from the geometry with justification from a geometric evaluation.

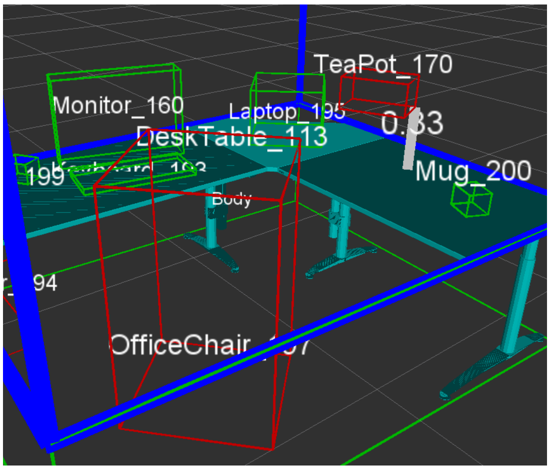
Unfortunately, both PostGIS and SFCGAL offer no 3D containment tests. Hence, we extended SFCGAL with such operators by using existing CGAL algorithms. The current implementation is limited to detect containment for a set of target points or a polyhedral body within a reference polyhedron. It enables SEMAP to evaluate 3D containment on all 3D bounding volumes, which are represented by polyhedral mesh data. Examples for SEMAP's 2D and 3D containment tests are presented in Figure 5.7 and 5.8.

Directional Operators

PostGIS natively provides a set of *directional operators* to identify if a geometry is `left-of`, `right-of`, `above-of` or `below-of` another geometry. They operate on the 2D axis-aligned bounding boxes of the query geometries only. Thus, they identify the directional relations with respect to the *extrinsic* global reference frame, but not based on the object's *intrinsic* reference frames, which limits their utility for robotic applications. Figure 5.8 (a) demonstrates this problem: the native operators can not infer that the *ConferenceTable* is `in-front-of` most of the *ConferenceChairs*, but `behind-of` *ConferenceChair116* as to the chair is facing away from the table. Another issue is that the operators neglect the third dimension, which makes it impossible to determine that the *TeaPot* in the depicted scene is `above-of` the *ConferenceTable* in a three-dimensional sense. To overcome these shortcomings and allow for spatial analysis using 3D directional relations and intrinsic reference frames, we implemented the projection-based and halfspace-based model, as proposed by Borrmann [46].



(a) 2D Directional Models



(b) 3D Directional Models

Figure 5.8: An application example of SEMAP's custom directional operators in 2D and 3D.

In the projection model, the faces of each object's bounding box are extruded to create six box geometries on top of every face. The extrusion's distance is determined by multiplying the object's extent in the respective dimension by a scaling factor. In the half-space model, six additional box geometries are created by first extending the bounding box faces along the two secondary axes before extruding along the primary axis. The extrusion's direction follows the conventions for object descriptions and both models are stored within an object description's set of abstraction models and transformed accordingly for each object instance. These additional box geometries can be used to evaluate 2D and 3D directional relations from the object's intrinsic point of view. The containment operator is used for a *strict* interpretation of directional relations, labeling only those object to be in the tested relation if they are completely within the projection space. For a *relaxed* interpretation, the intersection operator is used, which allows for partially included objects, too. The presented directional models are quite basic and could, if required, be exchanged with more elaborated models.

Figure 5.8 presents examples of these auxiliary geometries. In (a), the light red 2D box extending *ConferenceChair180*'s front, as well as the dark red box extending from *ConferenceChair116*'s back constitute projection geometries. These geometries now properly reflect the reference object's intrinsic viewpoint, e.g., *ConferenceChair180* is `behind-of` *ConferenceChair116*, whereas *ConferenceChair116* is `in-front-of` *ConferenceChair180*. (b) presents a 3D example for detecting directional relations. The blue box above the desk's top is used to detect objects that are above it, such as monitor, laptop, and mug.

By combining directional and topological operators, additional spatial relations can be identified. To evaluate the is-on relation, SEMAP pairs the strict 3D above-of operator with an additional distance constraint that rejects all target objects beyond a certain threshold distance, such that it can find all objects that are on another object. In Figure 5.8 (b), for example, the chair's bounding box violates the strict above-of relation relative to the desk's top projection, and the teapot's bounding box exceeds the distance threshold, because it was artificially placed way above the desk's surface. All other objects are correctly classified as being on the desk.

5.3.6 Query System

Querying SEMAP for environment knowledge is done by using the methods defined in SEMAP interface layer. This interface layer also handles the synchronization between SEMAP's spatial data base and the knowledge base.

The PostGIS back-end is queried using SQL, whereas the Apache JENA triplet store provides a SPARQL endpoint. Currently, there is no automatic synchronization between the two querying interfaces, such that the synchronization has to be explicitly triggered in the correct order.

First, the spatial database is triggered to evaluate binary spatial operators that identify a relation between two geometries, e.g., determine the distance between two objects. For this, a set of reference and target geometries must be assigned. To restrict the set of geometries in terms of their object classes, SEMAP relies on the semantic labels assigned in the PostGIS data base. This allows to impose semantic constraints during the spatial querying process. It is also possible to refer to specific objects by using their IDs directly. The type of spatial representation can be constrained, as well. SEMAP allows to use both complete body geometries, as well as the given abstractions in 2D and 3D. All geometries must obviously be drawn from the absolute view on the object instances. The semantic and geometric constraints are evaluated prior to filtering. An example is presented in Figure 5.9.

```
def get_distance_between_objects( call ):

if call.max_distance:
  distance = ST_3DMaxDistance(ref_geo.geometry, tar_geo.geometry)
else:
  distance = ST_3DDistance(ref_geo.geometry, tar_geo.geometry)

pairs = db().query( ref_obj.id, tar_obj.id, distance ).\
filter(
  ref_geo.id.in_(get_geo_ids(ref_obj, obj_const, geo_const)),
  tar_geo.id.in_(get_geo_ids(tar_obj, obj_const, geo_const)),
  ref_obj.id != tar_obj.id )

if call.sort_descending:
  result = pairs.order_by( desc( distance ) ).all()
else:
  result = pairs.order_by( distance ).all()
```

Figure 5.9: A code excerpt of SEMAP's distance measurement operator.

Once executed, the spatial query returns pairs of object IDs that satisfy the spatial relation tested for. Depending on the operator, additional information is returned as well, i.e., the respective distance between the objects. The spatial relations are now grounded in terms of a quantitative geometric analysis. Next, as they represent facts about spatial predicates holding between entities, they are accordingly asserted in the OWL-based A-box in JENA's triplet store. After the insertion it is possible to semantically query for spatial relations. In this case, the SEMAP ontology provides additional information about the entities and relations encountered in the environment's domain through ontological reasoning about the conceptual background knowledge in the T-box. Other types of inference, for example, using rule-based reasoning, can be used from here.

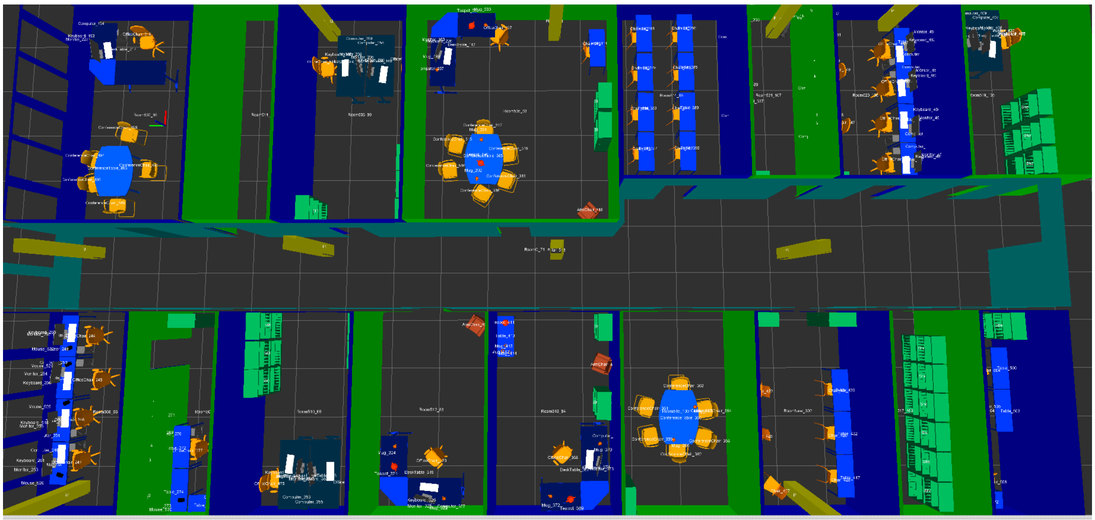


Figure 5.10: The created data set remodels an office environment at Osnabrück University.

5.3.7 Performance Evaluation

To evaluate the performance characteristics of spatial queries, we conducted a set of experiments. Since SEMAP makes no assumptions about the environment model's data source, it allows to build environments from sensor data, as well as from provided CAD models. For the following evaluation we constructed a reference data set from CAD data modeling the building of the Computer Science department at the University of Osnabrück depicted in Figure 5.10. The data set contains a total of 300 individual object instances that were created from 35 different reference models used as gemetric object descriptions. The objects are spread across 18 different rooms, which resemble real offices, computer labs and seminar rooms. Examples of how to generate similar maps from real sensor data are presented in Section 5.4.1.

Based on this data set, we conducted a set of test queries, to exhibit the performance of SEMAP's spatial querying and to differentiate different strategies for using spatial operators. All test where conducted on a Lenovo ThinkPad W530 with Intel Core i7-3940XM (4x 3.0 GHz, 8MB cache) and 8GB RAM.

First, we tested the execution of the containment operator using two different query types. The first type creates an inventory list for all rooms by performing a many-to-many strategy with instances of `Room` as reference set and an unrestricted target set. The second query type provides a full enumeration of all object pairs matching the `is-in` relation, by using a completely unconstrained many-to-many strategy. Both queries were performed using the strict containment operator in 2D and 3D and on different abstraction levels.

The results shown in Table 5.4 provide two insights. First, increasing the geometric abstraction level decreases query selectivity and vice versa. Testing, for example, 2D positions against the 2D bounding boxes returns more results than testing bounding boxes against each other. This is expected, since the latter is more restrictive. Evaluating against convex hulls is even more selective. The same holds for the comparison between queries executed in 2D and 3D, here evaluating in three dimensions is obviously more selective.

Dim.	Reference			Target			Num. Tests	Time/Test [s]	Total Time [s]	Num. Pairs
2D	Room	BB2D	18	All	POS2D	300	5400	0.000007	0.039337873	282
	Room	CH2D	18	All	POS2D	300	5400	0.000012	0.064801216	279
	Room	BB2D	18	All	BB2D	300	5400	0.000011	0.061866045	278
	Room	CH2D	18	All	CH2D	300	5400	0.000014	0.074759007	275
	All	BB2D	300	All	POS2D	300	90000	0.000003	0.269836902	439
	All	CH2D	300	All	POS2D	300	90000	0.000004	0.325406074	430
	All	BB2D	300	All	BB2D	300	90000	0.000002	0.203353166	363
	All	CH2D	300	All	CH2D	300	90000	0.000003	0.265438796	360
3D	Room	BB3D	18	All	BB3D	300	5400	0.033033	177.784672022	268
	Room	BB3D	18	All	POS3D	300	5400	0.016201	90.110987186	274
	All	BB3D	300	All	BB3D	300	90000	0.036631	3274.874104981	278

Table 5.4: Performance evaluation of the strict 3D containment operator.

Secondly, increased accuracy comes at computational cost and vice versa. Comparing the 2D data sets reveals that testing positions or bounding boxes against bounding boxes is considerably faster than testing against convex hulls. This is due to the fact that the necessary tests can be performed in constant time, since both geometries are of fixed size, whereas the geometric complexity of the convex hulls is usually higher and also varies among objects. In the 2D case, these differences are negligible, since each test only takes a couple of microseconds, so that even a large number of tests can be performed in reasonable time. The full enumeration of containment relations on 2D convex hulls was executed in 0.26 s for total of 90.000 tests.

For 3D spatial queries, however, the situation is different. Testing a single pair of 3D bounding boxes takes about 35 ms, which is reasonably fast for a small number of queries, but with an increased number of tests, the query time accumulates to minutes or more. The full enumeration of all containment relations using 3D bounding boxes, for example, took over 54 min. This tendency was expected, but PostGIS's performance on 3D geometries seems to leave room for optimization. Currently, the poor scaling of the 3D operators renders the direct evaluation of 3D spatial relation useless, especially in robotic applications that need near-realtime response. It is, however, possible, to narrow down the set of geometric tests, which addresses these performance problems, as described next.

5.3.8 Increasing Performance

The first strategy is to successively apply spatial operators, with an increasing level of selectivity and computational complexity. By applying coarser but quick to compute spatial tests, we narrow down the object pairs that need to be tested with computational expensive operators.

Figure 5.11 illustrates this strategy on a simple scene. Here, we want to test which objects shown in (a) are on the table. We could test for the 3D relations holding between all objects in this scene, directly or apply a 2D query as a filter query before. In (b), the convex hulls of the objects are shown. Querying for strict 2D containment reveals that mugs and tea pot are fully contained in the table's convex hull, while the chairs are not. Therefore, we can immediately rule out that the chairs may be on the table. The given 2D containment, however, may indicate that the target objects are either `in` or `on` the table or that 3D directional relations, such as `above-of` and `below-of`, may hold as well. We can therefore continue with testing for `is-on` based on the 3D bounding boxes.

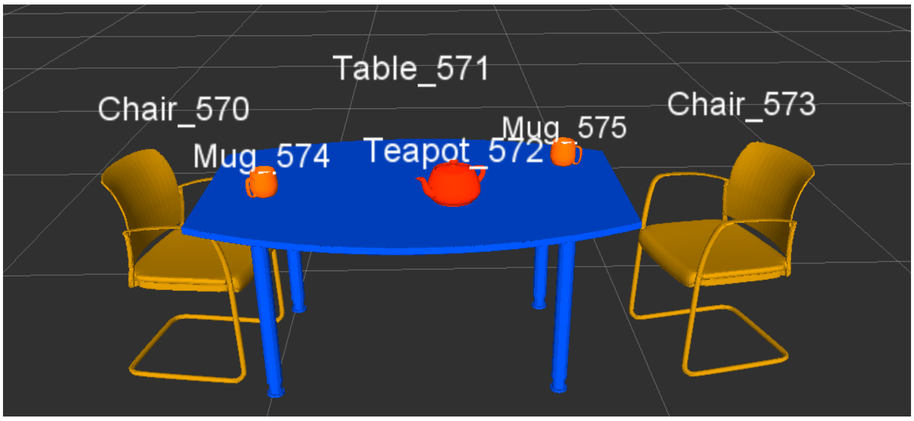
Testing for a 2D relation before applying the more complex 3D spatial operators, can effectively reduce the number of tests. To exemplify the advantages of this approach, we conducted the previously described experiment again, but used a 2D containment before testing for 3D containment. This significantly reduced the number of 3D tests from 90000 to 359, as well as the total runtime for testing from 54 min to 14 s. The results are shown in Table 5.5.

Similar performance increases can be produced for all other spatial operators when testing for intersections or directional relations. The actual run times vary from operator to operator and are dependent on the number and the complexity of the involved geometric tests. Using pre-queries to accelerate the query process is a strategy that can also be used across all spatial relations and on the different geometric representations and abstractions of an object. The choice is usually dependent on the application and always a trade-off between computational complexity and spatial accuracy. By default SEMAP's query interface already applies suitable 2D pre-queries, before executing 3D spatial queries, to allow online robotic applications.

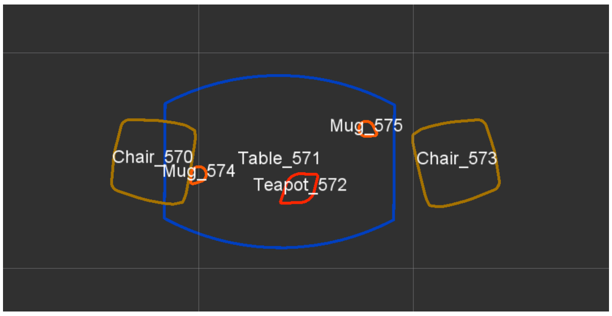
Another strategy to optimize query performance while keeping geometric accuracy, is decomposing objects into their individual parts before testing. By default, SEMAP's spatial query system performs object-to-object tests, using either the object's body geometry or a geometric abstraction that covers the entire object. Figure 5.11 shows two problems that arise: While geometric analysis on the actual object's body returns the most accurate evaluation, it is very costly, especially when the models are as detailed as the chairs in (a). An evaluation on the entire object's 3D orientated bounding boxes is faster, but may not provide the required level of detail. In (c), for example, an intersection would be found for *Chair570* against *Table571*, even

OP	**Reference**				**Target**				**Num.**	**Tests**	**Time [s]**	**Num. Pairs**
2D/3D	Room	BB2D	BB3D	18	All	BB2D	BB3D	300	5400	277	9.9509649270	268
2D/3D	Room	BB2D	BB3D	18	All	BB2D	PT3D	300	5400	277	4.837368965	277
2D/3D	All	BB2D	BB3D	300	All	BB2D	BB3D	300	90000	359	14.018936157	278
2D/3D	All	BB2D	BB3D	300	All	BB2D	PT3D	300	90000	359	6.91635704	281

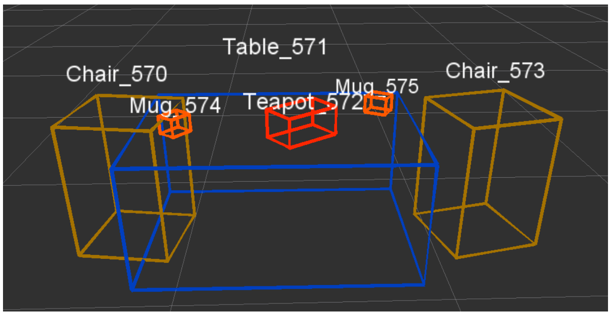
Table 5.5: Performance evaluation of the strict 3D containment operator using a 2D pre-query.



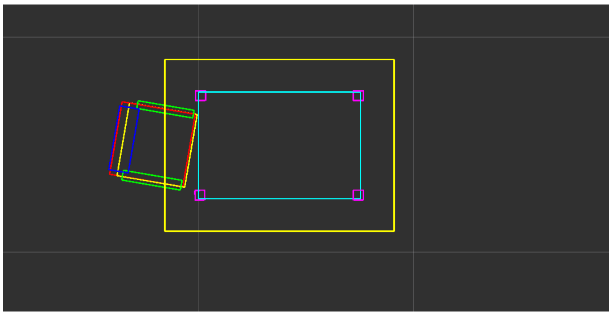
(a) 3D Scene



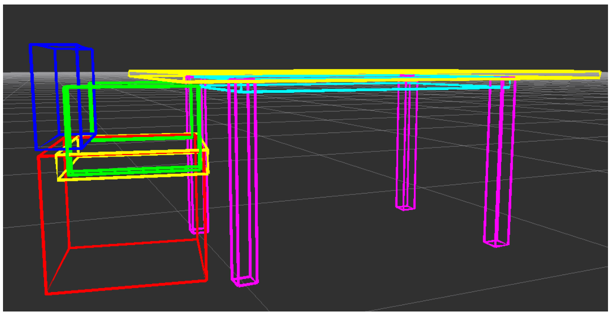
(b) 2D Convex Hulls



(c) 3D Bounding Boxes



(d) 2D Decomposition



(e) 3D Decomposition

Spatial Relation	**Geometry**	**Target**	**Reference**	**Result**
`is-in`	CH2D	Teapot572	Table571	True
`is-in`	BB3D	Teapot572	Table571	False
`is-on`	BB3D	Teapot572	Table571	True
`intersects`	GEO3D$_{body}$	Chair570	Table571	False
`intersects`	BB3D$_{body}$	Chair570	Table571	True
`intersects`	BB3D$_{links}$	Chair570	Table571	False

(f) Spatial Queries.

Figure 5.11: (a) shows a simple 3D environment, (b) the 2D convex hulls and (c) the 3D oriented bounding box of the entire object, whereas (d) and (e) show the oriented bounding boxes of the individual parts of the objects in 2D and 3D, respectively. (f) shows the evaluation of different spatial relations and their results.

though there is none between the object bodies. Since SEMAP supports composite objects and kinematic chains, it is further possible to query for spatial relations by considering the individual links of objects. By decomposing the objects into parts as shown in (e), we can significantly reduce the computational complexity and still determine the desired results, as (f) shows. To find cases where this object decomposition scheme is valuable, again a 2D query for intersecting object footprints can be used, before applying more complex operators. In (d), for example, the intersection of the entire object's 2D convex hulls can be used as an indicator that the links may need to be evaluated individually. Increasing the level of detail, e.g., by checking (d) the link's individual 2D bounding boxes, may then provide the information that a 3D test must be executed only for the table's surface against the chair's seat, legs and arm rests. Testing these links against each other using 3D bounding boxes, finally reveals that no links intersect and applying the directional operator shows that these parts of the chair are indeed below the table surface, as shown in (e).

We manually segmented the chair's CAD model and imported it into SEMAP via an URDF description. One could, however, add automatic object decomposition functionality to the framework, to use this strategy without additional manual effort. See [209] for a suitable approach.

Figure 5.12: Detection of furniture instances from RGB-D data using CAD reference models as presented in [108].

5.4 Practical Applications

This section demonstrates how to perform combined spatial and semantic queries with the SEMAP framework in order to support various applications that benefit from semantic environment data.

5.4.1 Map Generation from Sensor Data

For practical applications it is crucial that SEMAP is able to handle semantic information from a real mapping process on a mobile robot. Since the framework can handle all geometric data types supported by PostGIS, it is possible to add semantically annotated objects to SEMAP from different mapping approaches. Figure 5.12 shows results from an approach that uses CAD reference models for semantic classification in RGB-D data [108]. If CAD models are not available, surface reconstruction methods in combination with semantic classification based on planar constraints can be used to create a semantically annotated polygonal representation from incoming sensor data, as shown in Figure 5.13.

This is to illustrate that arbitrary annotated polygonal data – including appropriately converted octree representations – from actual robotic data can be fed into SEMAP instances. For this article, we tested SEMAP with data from the approaches presented above, but the integration of other reference data sets like NYU [189], Robo@Home [179] or others is clearly feasible after the implementation of suitable converters.

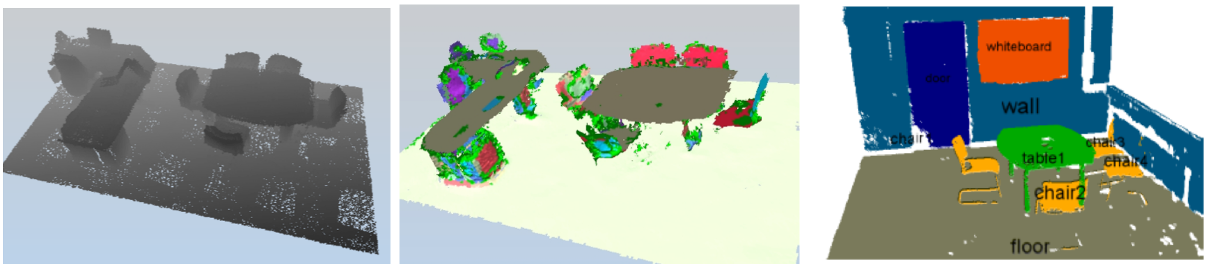


Figure 5.13: Semantic labeling of polygonal reconstructions from point cloud data (left) based on normal orientations and planar relations [152, 223] (center) to a SEMAP model (right).

5.4.2 Topological Structuring

Performing spatial and semantic analysis on the environment model can make information explicit that is otherwise only implicitly encoded in the data. The topology of an environment, for example, is covert in the spatial relations that hold between objects and can be revealed by applying spatial operators. The extraction of topological knowledge is a key feature of our semantic mapping framework and is of great benefit for path planning and exploration, especially during the initial map building process, when many spatial relations need to be grounded at once.

To bootstrap assertions on topology underlying our test data set, we used queries like the ones evaluated in Section 5.3.7, to create an inventory of all rooms and used it to insert the objects into SEMAP's knowledge base afterwards. The obtained spatial predicates were then used to restructure the environment's transformation tree to reflect the topological relations between the objects. We use the containment relation, to bind the objects found in each room to the reference frames of the respective room. See Figure 5.14 for an example.

A subsequent query identified all objects that are `on` objects of type `Table`. The results were also used to bind the target objects to their parent's frame. This step is illustrated in Figure 5.15: (a) shows a the transformation tree of a single room before and (b) after the objects are bound to their supporting tables. (c) shows a close-up of single table. Since the redirection of a reference frame is negligible, the run time of a batch-wise topological restructuring compares to the performance of the strict containment operator as shown in Table 5.5. Of course the same

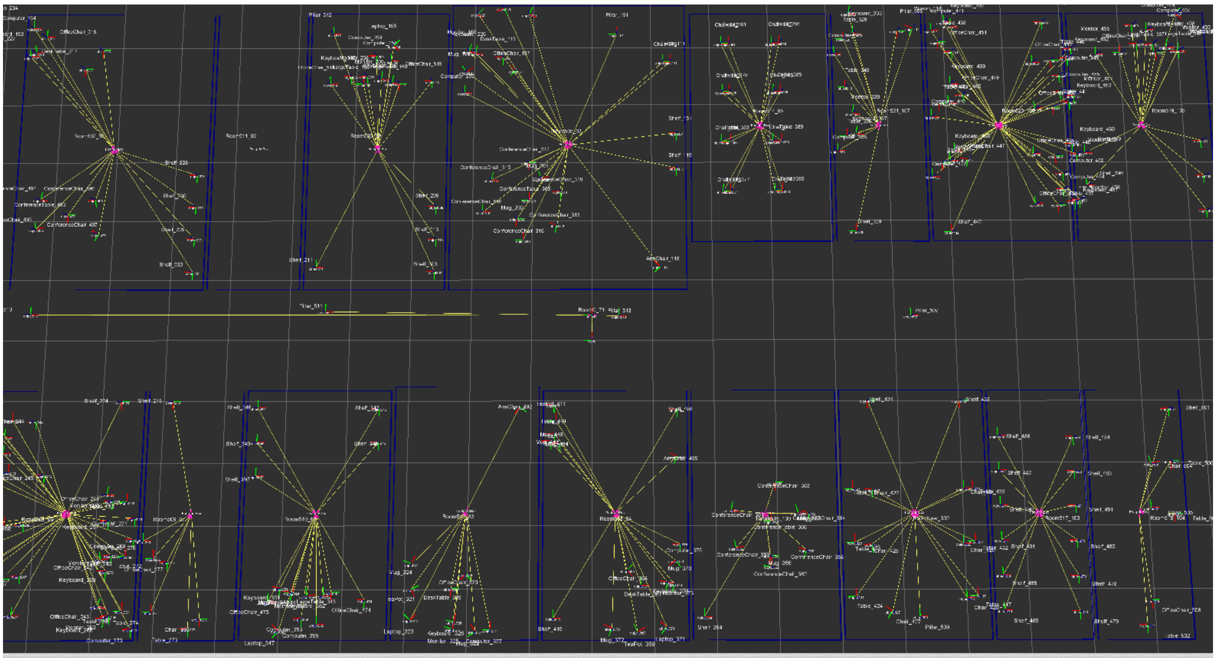


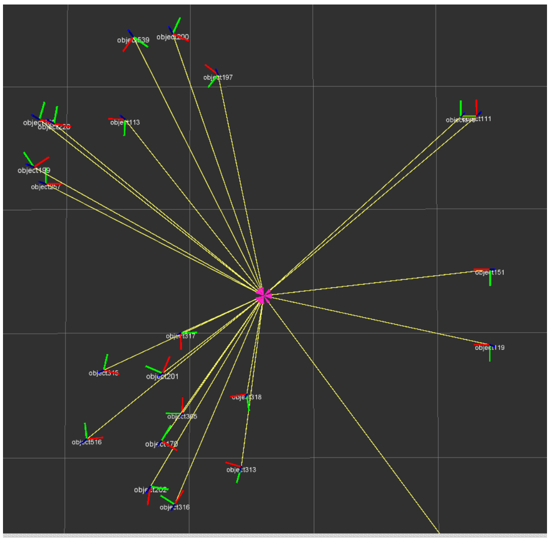
Figure 5.14: Applying containment queries on a global scale allows to structure environments topologically. Here, the 2D bounding boxes of all `Rooms` were used to structure the environment displayed in Figure 5.10.

procedure can be applied for other common objects with surfaces, as well as parts of objects, e.g., the boards of a shelf.

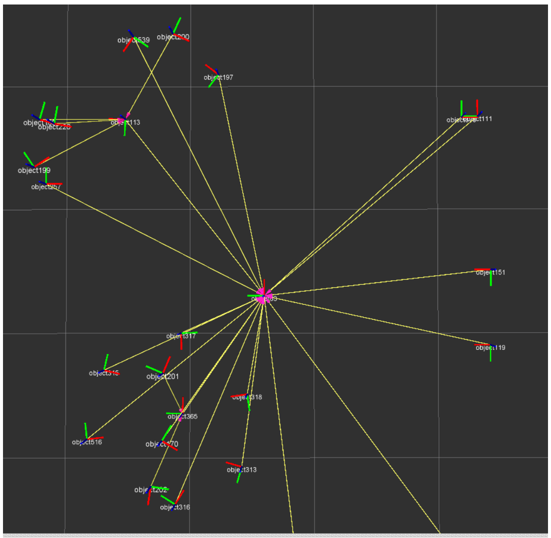
Applying topological restructuring of the relative transformations brings several benefits: First, objects move together with their topological parent, e.g., a mug bound to a table moves if the table is moved. Second, the explicitly encoded topology can be queried directly from the transformation tree, which is considerably faster. An example: a spatial containment query to evaluate the objects within `Room505` took about 0.94 s, whereas retrieving the same inventory list from the transformation tree after the environment was topologically restructured took merely 0.0025 s. This significant drop in retrieval time is owed to the fact that a relational database lookup is considerably faster than spatial queries, as no geometric analysis is involved. Third, all explicit relations can be returned as a topological graph that can serve as input for topological navigation, without taking the detour over the knowledge base.

5.4.3 Object and Scene Classification

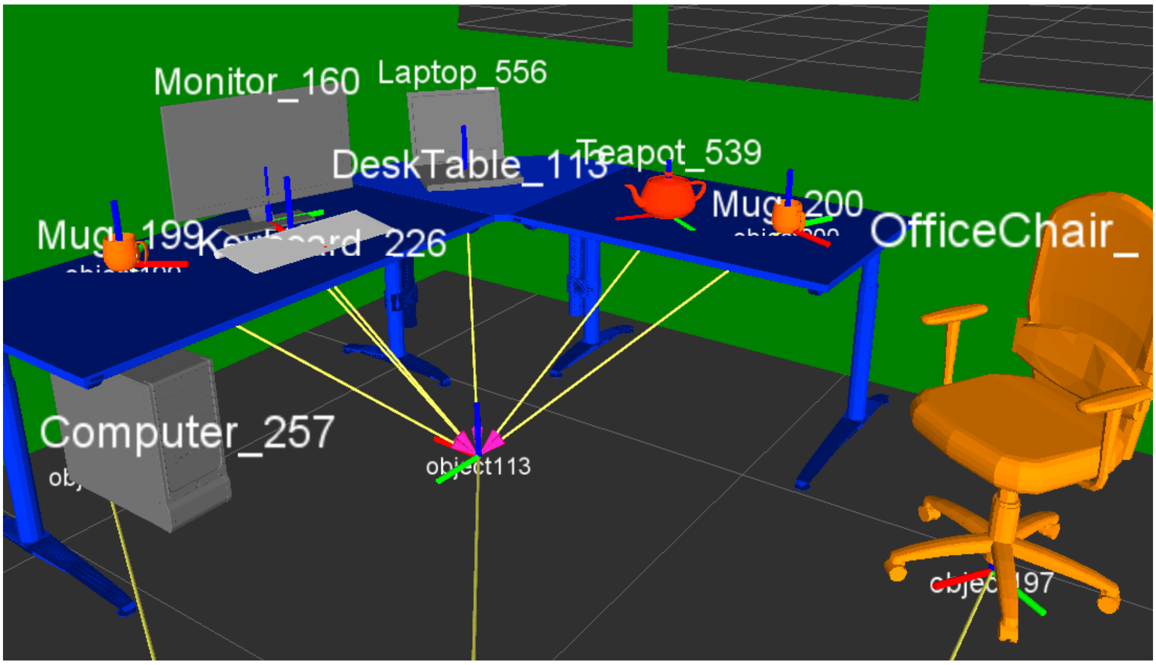
Topological analysis can also be the source of further insight into the environment's semantics. For instance it is possible to classify groups of objects into high-level aggregates or discriminate between type of rooms depending on their inventory. This requires suitable background knowledge and rules that discriminate object properties, assign additional attributes or create new entities.



(a) Topologically Structured By `in`



(b) Topologically Structured By `in` and `on`



(c) 3D Close-Up

Figure 5.15: (a) shows the transformation tree of a single room structured by evaluating the spatial relation `is-in`. (b) shows the same scene structured by additionally evaluating the `is-on` relation. (c) shows a table in close-up to illustrate how objects on the table are bound to the table's reference frame.

```
?room rdf:type office:Room
?room semap:hasObjectModel ?room_obj
?room_obj semap:hasConvexHull2D ?room_abstr_ch2D
?desk rdf:type office:DeskTable
?desk semap:hasObjectModel ?desk_obj
?desk semap:hasConvexHull2D ?desk_abstr_ch2D
?desk_abstr_ch2D semap:isIn2D ?room_abstr_l2D
==>
?room rdf:type office:Office
```

Figure 5.16: A rule classifying a room as an office, due to a particular type of table in it.

Such rules can be implemented, for example, by adding an SRWL rule interpreter to the Apache JENA back end. Figure 5.16 shows a simple classification rule that uses the concepts defined in our office domain, as well as the grounded spatial relations, to specify that an instance of type `Room` is actually of type `Office`, due to the fact that it contains a specific type of table, namely a `DeskTable`. The scene in Figure 5.15 (c) would qualify for this rule-based classification. This type of reasoning is used as a key component in [108].

In a similar style, other room types could be distinguished from each other based on their contents. For the scene shown in Figure 5.17, one could identify instances of `DesktopComputer` and `Monitor` within a narrow search radius around an instance of `Table` and group those into a new object entity of type `Workplace`. Afterwards aditional queries over the number of workspaces contained in a room could be used to differentiate between `ComputerLabs` and `Offices`.

5.4.4 Object Retrieval

To search for objects based on spatial and semantic criteria is an asset in many robotic activities, ranging from task planning and object manipulation to human-robot interaction. SEMAP's query system can be of help in all such applications.

Imagine the robot shown in Figure 5.18 (a) is asked to perform fetch-delivery tasks, e.g., to bring the operator *his* coffee mug. To solve this task, the robot is challenged to find out possible target mugs within the environment and identify the correct instance. Hence it has to answer various basic queries about the environment. A query like *Q1* in Figure 5.18 (b) will provide a set of potential targets and the rooms they are in. To narrow down this selection to the actual target, additional information is needed. However, a query about topological relations, such as *Q2*, may enable the robot to formulate natural questions, e.g., "Do you mean the mug *on* the desk, *right of* the laptop?". A likely response could be: "No, mine is *on* the `Shelf`.". This additional information allows to filter the results of *Q1* down to a single instance, namely `Mug3` and thus yields a distinct target for the robot. If the robot's next task is to serve tea, it can issue a query like *Q3* that directly identifies the most suitable target, the `Teapot`, and immediately retrieves its pose and relative geometries to guide the navigation, grasp planning, and object manipulation. Note that the latter query can be enriched with robot-dependent information, such as the maximal viable bounding volume to fit the robot's gripper, in order to extract only suitable matches.

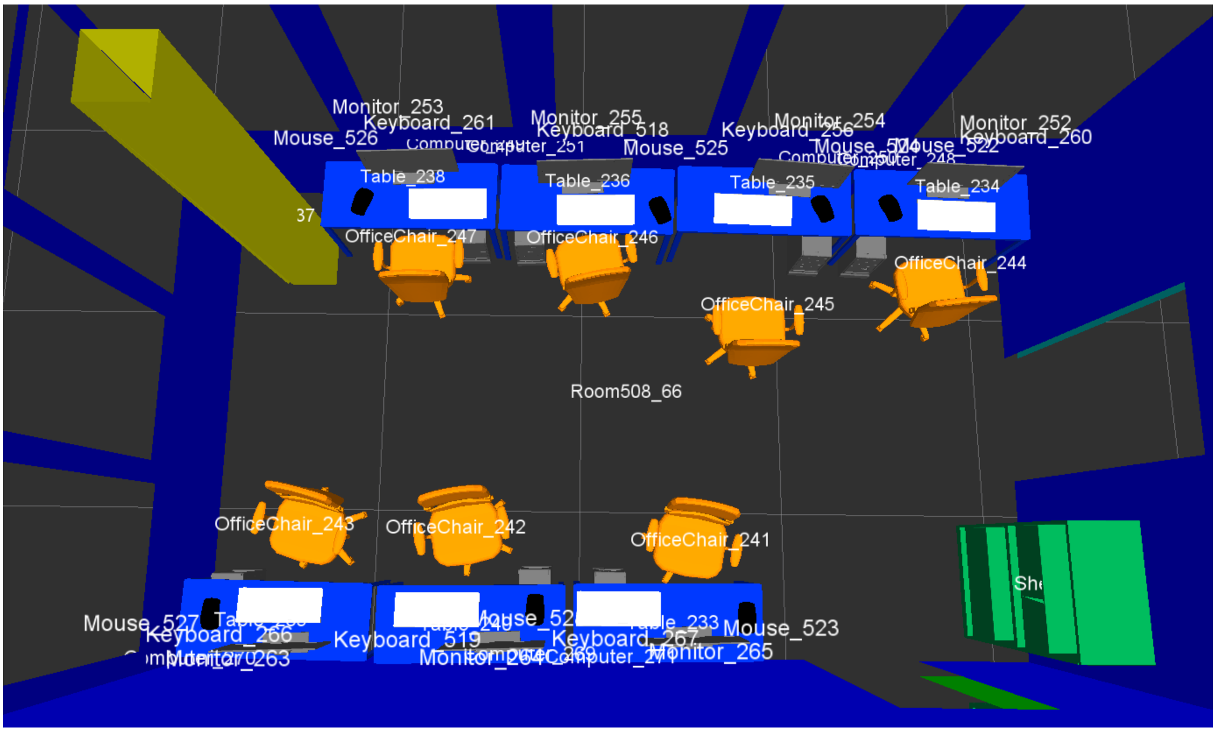


Figure 5.17: A SEMAP scene of a computer lab room with multiple workspaces, each consisting of a table, desktop computer and monitor.

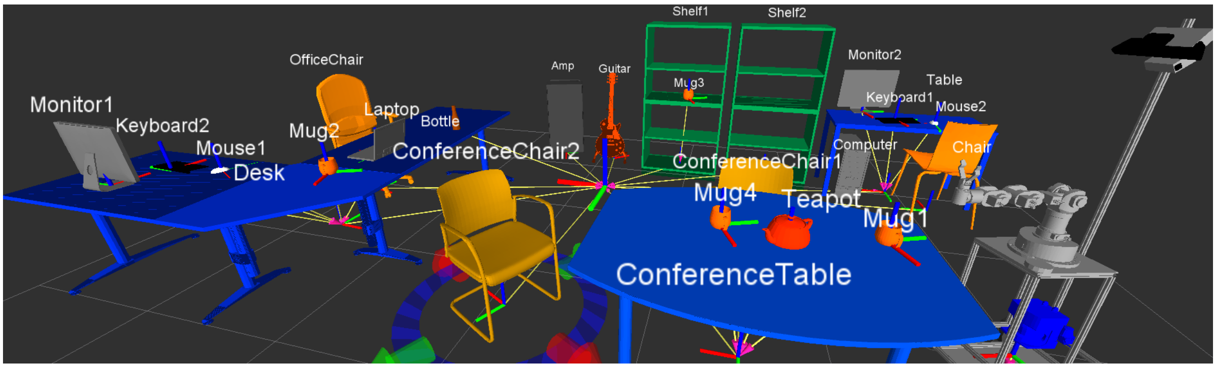
5.4.5 Environment Awareness & Dynamic Map Updates

Obviously the just-mentioned object retrieval queries are only useful if dynamic changes in the environment are continuously detected and incorporated into SEMAP's representation.

To detect dynamic changes in the environment, we proceeded as follows. We first created a module that implements environment awareness for our mobile robot and informs its object recognition module about entities that can and should be tracked. It identifies these objects of interest (e.g., tables and chairs) within a search radius around the robot, as depicted in Figure 5.19 (a). It uses a continuously executed range query using SEMAP's distance operator on a parameterizable set of objects. This informs the robot about the it need to track and check whether they are still present or not.

To this end, we use the currently stored object locations to navigate to the nearest object. Then we use parts of our map-building pipeline for object recognition. We hereby rely on a CAD matching approach, similar to the one presented in [108]. The necessary CAD models are provided by SEMAP directly. If the object is recognized at the given location, we use the returned estimate on the object pose, as an update to SEMAP's environment model. Once the object is updated, a spatial relation extraction query is automatically triggered to inform the knowledge base about potential changes in the environment topology. If the object is not found at the given location, we retract the entity from the environment model.

We also use the strategies presented above, to topologically re-structure the environments



(a) An office environment modelled with the SEMAP framework.

Query / Result
Q1 Return all mugs and the objects (parts) they are on.
R1 `Mug2` on `Desk`; `Mug1` and `Mug4` on `ConferenceTable`, `Mug3` on `Shelf1-Board3`
Q2 Which relations holds for `Mug200` with respect to desk and laptop?
R2 `Mug2` *is-on* `Desk`, *right-of* `Laptop`
Q3 Return pose and geometry of a (graspable) teapot.
R3 `Teapot`, $Pose_{27}$, $RelativeGeometry_{27}$

(b) Several query types that let SEMAP support different robotic applications.

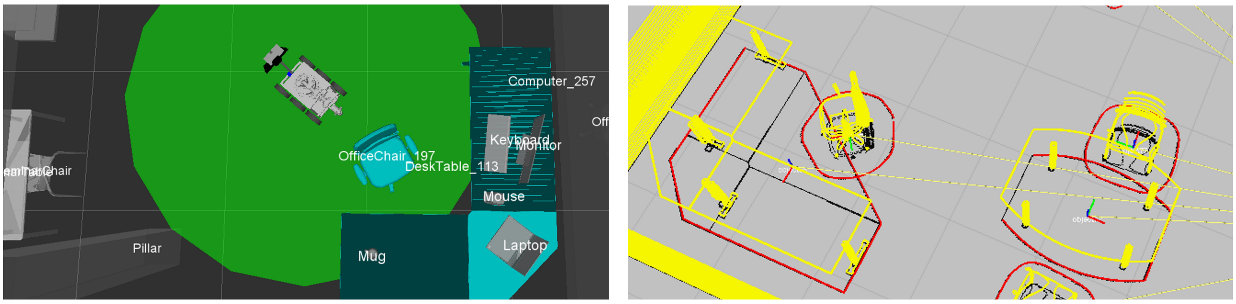
Figure 5.18: An exemplary office environment and questions referring to objects in it that may come up in fetch-and-delivery tasks for a service robot in such a place.

model after every map update, ie., when an object has moved or a new object was created. To correctly insert a new object, the run time is around 0.3 s on average, which indicates that the environment topology can be maintained with every change on our mobile robot.

Currently, we can not track the articulation of environment entities, since our perception pipeline is limited to detecting rigid objects, yet we were able to test SEMAP's ability to represent the dynamics of articulated objects by using our mobile robot itself as a test sample. We imported the robot's URDF model to create an articulated object entity within SEMAP. Next, we continuously fed the robot's pose estimate and joint states into the environment model, to align the robot's SEMAP model with the current world state. In doing so, we are able to query for spatial relations between the robot's links and the environment, i.e., we were able to infer that the robot's gripper is over a desk during the execution of an object manipulation task, as shown in Figure 5.20.

5.4.6 Navigation Map Extraction

SEMAP represents a model of the robot's environment, from which multiple applications can retrieve task-specific environment data on demand, rather then maintaining several different



(a) Environment Awareness **(b)** Robot Navigation Extraction

Figure 5.19: Two applications that make use of dynamic data extraction from SEMAP's environment model: (a) filters the robot's environment for relevant objects, whereas (b) extracts a map for navigation.

semantic map representations simultaneously. In this sense, SEMAP is a hybrid map, but with the additional freedom of deciding at run time which set of map representations suits the given applications best.

As an example for the extraction of task-specific maps, we implemented a module that extracts grid maps for localization and navigation from SEMAP's database. It queries the environment's absolute geometries and dissects these into multiple horizontal slices, which are then used to create a 2D projection of the environment's 3D geometry. Converted into an occupancy grid map, this projection is made compliant with the standard algorithms for robot navigation in ROS. A parameterizable set of rooms and objects is used to tailor the extraction processes to the robot-dependent demands of the application. By default, we create the navigation maps for the entire floor on which a robot operates, including all geometries along the robot's height. We also augment the grid maps to restrict the robot from areas in which it may disturb humans. For this we use SEMAP's semantic knowledge to identify all chairs and then add safety buffers to their geometries using additional PostGIS operators for applying a spatial padding. Figure 5.19 (b) shows the map extraction process. The horizontal slices through the environment geometry are shown in yellow, the convex hulls of the blocked objects in red, the resulting occupancy grid is shown in black. Note how the projected boundary around the chairs is larger than their spatial footprint, due to the semantically-augmented spatial padding.

By registering to dynamic changes in SEMAP's database, we avoid the inconsistencies that may result in robotic systems when 2D navigation is decoupled from 3D data processing. It is further possible to work with multiple instances of the map extraction module on a single SEMAP model. These can either provide multiple robots with customized maps or a support a single robot's 3D navigation using a stack of 2D grid maps, like in Figure 5.20. A detailed description of the map extraction module used there is presented in [78].

5.4.7 Changing the Application Domain

So far, all examples where confined to service robotic tasks in an office domain. To clarify that SEMAP defines a domain-independent framework for constructing environment models, we

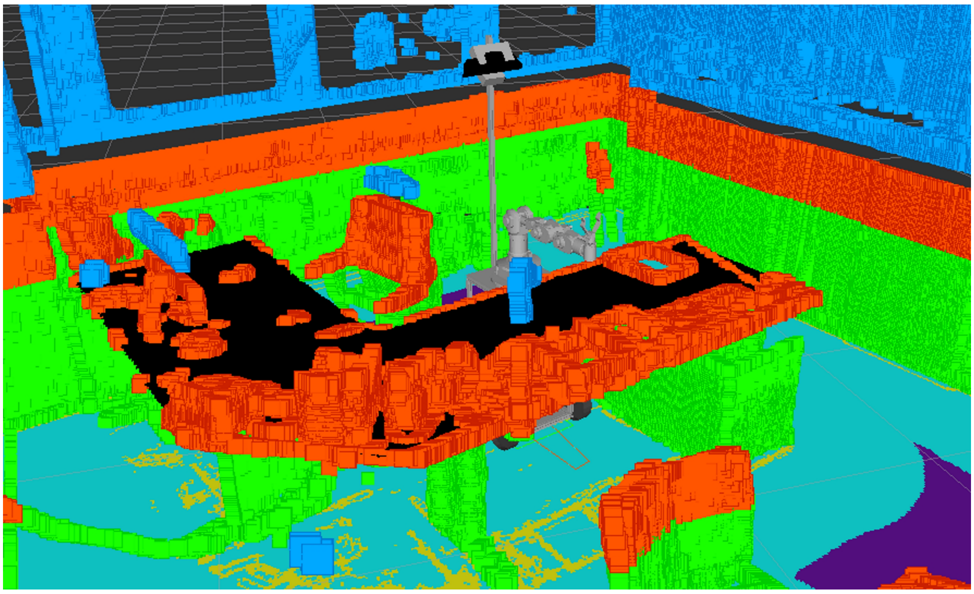


Figure 5.20: To query for the spatial relations between the robot and environment entities, the SEMAP model of our mobile robot is continually updated using its pose estimate and its joint states to describe its articulation state.

applied it in an entirely different application domain. To achieve this, SEMAP's core ontology has to be paired with a suitable ontology for the new application, such that domain-specific knowledge can be represented. The underlying representations and reasoning mechanisms remain the same.

In recent work, we applied SEMAP in an agricultural context [79]. For this, we simply replaced the office ontology used throughout, with a new domain model. Figure 5.21 shows this ontology, which describes entities in agricultural environments, such as fields, farms and tractors.

We generated an environment representation based on this model, by importing URDF models of agricultural machines, as well as a set of fields, represented by using polygonal boundaries and silo facilities, using 3D CAD models. Next, we use recorded telemetry data from a real harvesting campaign, to replay real machine movements between a field and a silo facility in our SEMAP model. We then used SEMAP's spatial and semantic reasoning capabilities, to detect spatial relations between the agricultural machines and their environment, to gain insight into

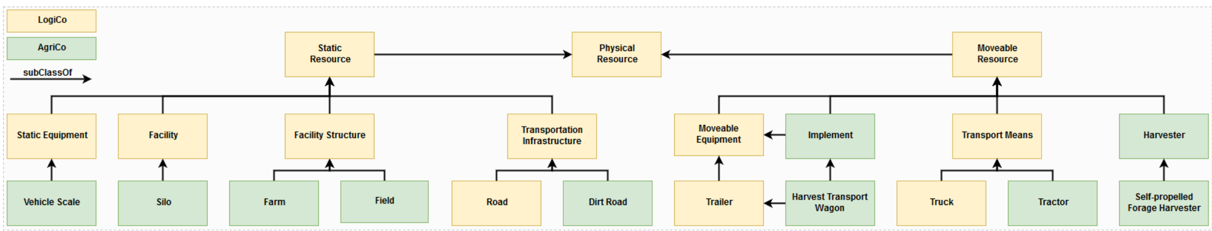


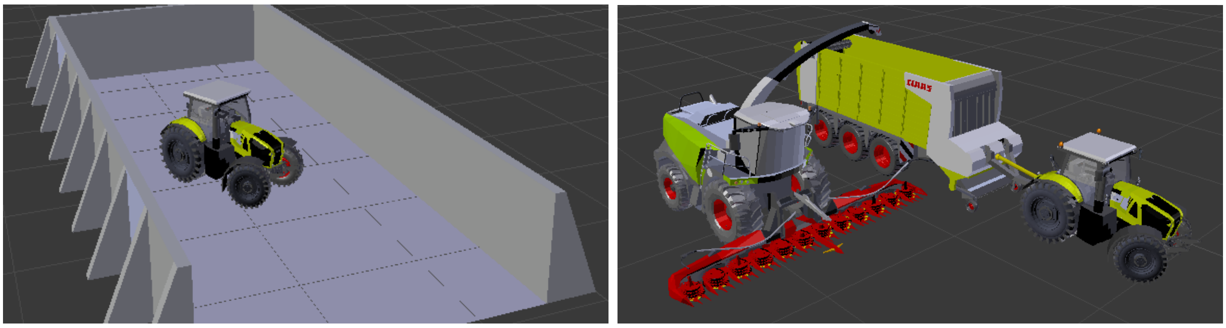
Figure 5.21: Excerpts of an agricultural-specific domain model added to SEMAP.

the agricultural process underlying the machine activities.

For example, we continuously identified the topological relations that hold between a movable entity, such as a tractor, and the set of agricultural facilities, namely the fields, farms and silo facilities. We used the 2D position abstraction of each tractor and harvester to test for containment against the facilities 2D polygonal boundaries. We use the positive results for grounding a generic predicate `isAt`, as well as specific predicates defined in the agricultural ontology, such as `inField`, `onFarm` and `atSilo`.

We used SEMAP to reason about more complex spatial relations, too. For example, we combined several basic directional relations about a harvester and a transport vehicle, to construct the domain-specific relation of describing that both vehicles are correctly positioned for overloading harvested goods. In a situation like the one shown in Figure 5.22 (b), we started with a range query to detect if the transport vehicle is close enough to the harvester. If so, we tested whether the trailer is left of the harvester (or right – depending on the orientation of the overloading boom) and if the harvester's overloading boom is over the trailer. If so, the relation `positionedForOverloading` is inferred to hold between both machines.

This is valuable information about the underlying agricultural process, which was previously covert in the telemetry data of both machines, but due to SEMAP's spatio-semantic processing is now explicitly available within SEMAP's KB, where it can be used for further processing, such as rule-based reasoning and eventually for planning and controlling the agricultural machines.



(a) Localization of a tractor in a silo **(b)** Detecting an overloading situation.

Figure 5.22: We used telemetry data from an actual agricultural machines to dynamically synchronize an environment model in SEMAP. Using the spatio-semantic query interface, we were able to topologically localize machines within agricultural facilites (fields, farms and silos) and to identify the correct positioning of two machines for overloading harvested goods.

5.5 Summary and Discussion

In this article we presented a semantic map representation framework called SEMAP that uses spatial database technology, to effectively ground qualitative spatial relations in order to make them available for knowledge-based reasoning. We extended PostGIS to support spatial queries in 3D and used its quantitative geometric analysis, to derived qualitative facts about the spatial relations of entities within an environment model. To bridge between geometric and semantic representations, we linked the entities from the geometric storage in the PostGIS database to concepts in an ontology modeled in OWL and implemented an data management and query interface that inserts these spatial predicates into a dedicated knowledge base, represented through Apache Jena, which allows for subsequent qualitative spatial reasoning on a symbolic level. To effectively realize the evaluation of geometric tests for complex geometries, we integrated suitable geometric abstractions into SEMAP's spatial model and added automatic optimizations to its querying strategies, such that time consuming tests are only executed when needed.

We presented the database schema to store static and articulated objects within the spatial database and the core ontology that is used to represent their semantic counterparts in the knowledge base. The separation between geometric core concepts and application domain in the ontology allows to use the proposed framework in different contexts. We demonstrated the basic functionalities of SEMAP in an office domain. These application examples showed, how the current implementation is able to utilize the spatial analysis capabilities in classic tasks of mobile robotics, like map building, scene classification, object retrieval and navigation. To demonstrate that the framework can be easily adapted to represent different semantic contexts, we switched to an agricultural domain model. In this application example, we used SEMAP to detect overloading positions in an harvesting process, based on recorded machine telemetry and thus provided valuable insight into a real-world application. For changing the application domain, we simply substituted the underlying domain ontology, while re-using SEMAP's core ontology and its PostGIS database for representing and querying geometric environment data.

The main drawback of the current implementation is that the linking between geometric models and qualitative knowledge has to be maintained via the query interface. Currently, we trigger all relevant updates manually to ensure that derived information from the database is inserted in the knowledge base. This is an issue concerning performance and data redundancy, and is also inconvenient during application development. To solve this problem, a formal query language that includes querying over qualitative spatial relations directly could be used and integrated into SEMAP's query system. With such a formalism, it should be possible to detect whether relations are already qualified or not to call the respective spatial operators only if needed. A candidate for such a formalism could be GeoSPARQL, which we indent to investigate in future work.

Another issue is the performance of the spatial database back end. Even though GIS technology provides spatial operators off-the-shelf, their 3D spatial representations and geometric processing lacks the efficiency required for real time processing. Although we tried to minimize the query times, some queries produce significant latency which may lead to data loss when the environment model is updated with high frequency, e.g., when telemetry information from actual machines is analyzed. A possible solution would to integreate a optimized spatial back tailored specifically for 3D data. To improve qualitative spatial reasoning, it would be beneficial to integrate a dedicated qualitative spatial reasoning system, like SparQ [228] in addition to the geometric analysis based on PostGIS and CGAL. It will be necessary to evaluate which calculi are suitable and whether the current set of spatial operators supports the chosen calculi or not.

Currently, SEMAP can only handle a most likelihood model. It would be desirable to combine the strength of the current implementation with probabilistic methods to further enrich the stored and derived knowledge. Additionally, handling the histories of objects would be beneficial to track the positions of objects over time to support anchoring processes. These properties should be relatively easy to implement in terms of the used database, but making these information usable for knowledge based reasoning is an open issue and will definitely require to redefine the structure of our semantic mapping framework.

In spite of these conceptional and implementational issues, the general approach to integrate a spatial database into semantic maps was proven to be beneficial and the SEMAP framework provides a functional proof-of-concept. Having operators for quantitative spatial analysis readily available in the semantic map's representation helps solving the qualification problem of spatial relations and effectively supports further spatial reasoning in robotics. Placed in a processing chain where the data is pre-processed, e.g., using stream processing and probabilistic approaches, SEMAP in its current state can already solve a number of relevant problems in semantic mapping as the presented application examples clearly demonstrate.

Chapter 6

A Spatio-Semantic Approach to Reasoning about Agricultural Processes

Henning Deeken	Knowledge-Based Systems Group, Osnabrück University, Osnabrück, Germany
	CLAAS E-Systems GmbH, Dissen am Teutoburger Wald, Germany
Thomas Wiemann	Knowledge-Based Systems Group, Osnabrück University, Osnabrück, Germany
Joachim Hertzberg	Knowledge-Based Systems Group, Osnabrück University, Osnabrück, Germany
	DFKI Robotics Innovation Center, Osnabrück Branch, Osnabrück, Germany

Abstract

Digitization of agricultural processes is advancing fast as telemetry data from the involved machines becomes more and more available. Current approaches commonly have a machine-centric view that does not account for machine-machine or machine-environment relations. In this paper we demonstrate how to model such relations in the generic semantic mapping framework SEMAP. We describe how SEMAP's core ontology is extended to represent knowledge about the involved machines and facilities in a typical agricultural domain. In the framework we combine different information layers – semantically annotated spatial data, semantic background knowledge and incoming sensor data – to derive qualitative spatial facts and continuously track them to generate process states and events about the ongoing logistic process of a harvesting campaign, which adds to an increased process understanding.

6.1 Introduction

Digitization of agricultural processes currently concentrates on recording and processing telemetry data from individual machines to support precision farming. This implicitly leads to a machine-centric view on the ongoing processes. But many agricultural processes are complex, cooperative orchestrations of multiple machines. Automatic decision support in harvesting campaigns is still limited in assistance systems, as representations of cooperative agricultural processes and tools to analyze inter-machine relations are mostly missing.

Information on the whole process can not be derived from a single machine's telemetry data, but is covert in the combined telemetry of multiple machines. To embed this abstract data from different machines in the context of the ongoing process, machine data has to be fused with additional knowledge and information about the environment and the process itself. Most importantly, symbolic representations of the spatial relations between agricultural machines and their environment are needed to identify and monitor process states and associated events. Analyzing the geo-location of individual machines and processing of spatial relations between them is therefore a valuable contribution to automated process managing in agriculture. Modern agricultural machines already provide a geo-referenced stream of telemetry data, based on RTK-GPS. The positional data is often used to inspect the containment of machines in polygonal boundaries representing fields and farms, to spatially locate machines at those facilities. Such a quantitative, geometric analysis already extracts a lot of relevant information, but does not account for qualitative relations between the machines and facilities, nor does it enable knowledge representation and reasoning on a semantic level.

Representing such spatial relations in terms of a well-defined semantic terminology allows to infer complex facts, built up from basic spatial relations to take a process-centric view on harvesting campaigns. This requires a machine-readable environment model that can be paired with geo-referenced telemetry-data from agricultural machines to geo-localize individual machines and derive spatial relations between machines and their environment, respectively. To meet these requirements, we use the semantic mapping framework SEMAP [3] to represent an agricultural domain. We show how to create a semantic environment model for agricultural environments and machines and how to connect it to the underlying geometric model. We illustrate how to ground qualitative spatial relations between a static environment and a set of dynamic vehicles with SEMAP. We further extended this ontological model which represents the activities and events of a harvesting operation, to enable an event-based tracing of the process.

In an application example, we replay telemetry of a harvesting campaign to continuously update the spatio-semantic environment model to derive symbolic facts about the ongoing process. Via rule-based inference we analyze the domain-specific spatial relations of a maize harvesting campaign to detect events such as the correct positioning of a transport vehicle next to the harvester for loading.

6.2 Related Work

State of the art solutions in digital agriculture allow to record and process telemetry data of agricultural machines like position, velocity, and internal parameters like fuel consumption or mass throughput [202]. This data is used in precision farming to optimize the application of fertilizers or herbicides, and collected in farm management information systems to aggregate telemetry data to analyze the performance of agricultural machines [164]. They also help to plan agricultural operations by maintaining information about crop rotations [85] or by creating field boundaries and sub-plots based on GPS data [137] to support the application of fertilizers and herbicides tillage strategies [197]. Automated scheduling of entire harvesting campaigns is also possible [12]. Usually, these solutions operate on centralized systems with web-based front ends [121]. This often causes severe latencies due to connectivity issues in remote or rural areas [142].

Fleet overview applications inform the operators about an on-going harvest operation by exchanging telemetry information between machines in real time and display vehicle positions on a static 2D map. Process-related decision making is still completely in the operator's hands, as these assistance systems do not provide a context-dependent and process-oriented analysis. To automatically detect relevant situations that give insight into the agricultural process – e.g., an empty transport vehicle arriving at the field ready for loading – is a key feature to increase process transparency, which is necessary for improving agricultural efficiency through more process-oriented decision support systems.

To solve these problems, existing approaches from semantic mapping in robotics can be transferred to this application domain. Semantic maps are representations that in addition to spatial data provide assignments to known concepts for the mapped entities, such that semantic background knowledge can be used to reason about the environment [152]. Recent advances in semantic mapping are concerned with constructing general models of multi-modal environment data that can be flexibly queried for task-specific data in individual applications, see [128] for an overview.

Being able to analyze spatial relations in terms of qualitative predicates is important in data retrieval and reasoning. To fully utilize qualitative spatial reasoning, it is necessary to derive qualitative symbolic data from quantitative metric information. In [228], Wolter and Wallgrün pointed out that this process of qualification is essential for qualitative spatial reasoning in practical applications, but still rarely seen. The lack of qualification is also apparent when working with semantic maps. Tools for performing spatial analysis on quantitative metric data are also seldom used in semantic mapping. In our previous work [3], we showed the advantages of maintaining environment data in form of a generalized and persistent model, from which task-specific semantic maps can be extracted, rather than maintaining and aligning several different layers of semantic, geometric and topological information in parallel. We proposed to pair spatial databases and declarative knowledge bases to combine ontological and logical rule-based inference with spatial querying and analysis capabilities and called it the semantic mapping framework SEMAP.

In this paper, we integrate an ontology for agricultural processes into SEMAP to make knowledge about harvesting campaigns accessible for automatic analysis. We use this knowledge together with SEMAP's spatial reasoning capabilities to recognize relevant events in an maize harvesting process. In the presented experiment we were able to detect the correct positioning of an transport vehicle ready for loading based on recorded telemetry in a real life harvesting campaign.

6.3 The SEMAP Framework

The SEMAP framework is designed to represent and manage spatio-semantic environment data. Its purpose is to provide information about the objects and the environment in a specific application domain. It connects conceptual knowledge about the environment and factual

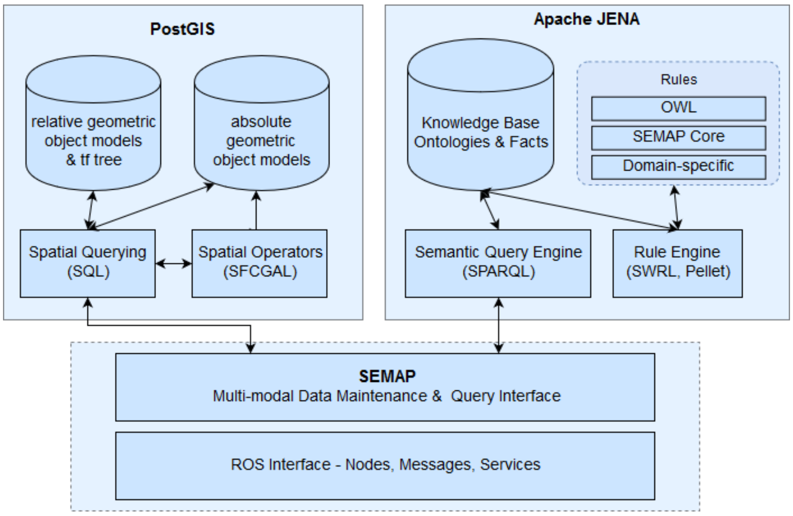


Figure 6.1: SEMAP's architecture features a spatial database and a knowledge base system, which are combined by a multi-modal querying interface.

knowledge about present object instances with their geometric representations to hold a combined spatio-semantic model that allows spatial analysis as well as semantic inference. To manage the fundamentally different structure of semantic and spatial information, SEMAP internally separates environment data into two dedicated databases to ensure optimized performance for each data modality especially in terms of data storage and retrieval. An outline of SEMAP's internal structure is given in Figure 6.1. The semantic part is represented by a knowledge base system component (KB) that is based on description logics with the obligatory separation into terminological and asserted knowledge. The environment's conceptual model and facts about the environment are represented in the Web Ontology Language (OWL) [24] and maintained in Apache JENA, which provides inference for ontological and rule-based reasoning as well as the capability to query the stored knowledge. The spatial part is a dedicated spatial database system (DB) that stores geometric primitives, and provides operators for quantitative spatial analysis and spatial querying. It is implemented as an extension to PostGIS using the SFCGAL plugin to create custom spatial operators, especially for detecting 3D spatial relations.

The framework's strength lies in combining both query systems to support combined queries with semantic and spatial aspects like *"Is there a computer in this room?"*, *"Which mug is (currently) the closest to the robot?"* or *"Is there a keyboard in front of the monitor?"*. In such queries, SEMAP utilizes the DB's spatial operators to ground qualitative spatial relations that are only stored implicitly in the geometric environment representation. Such relations are automatically inserted into the KB as facts for further inference. This approach enables rule-based reasoning and to construct complex spatial queries based on simpler deductions. This multi-modal query interface is advantageous in real-world applications, as it allows to answer complex questions about the positions, relations and roles of the stored objects in a natural way. The framework's core components are designed to be domain-independent, yet extensible with domain-specific semantic models, rule-sets and geometries. A more detailed description of the SEMAP framework and its spatial querying capabilities is given in [3].

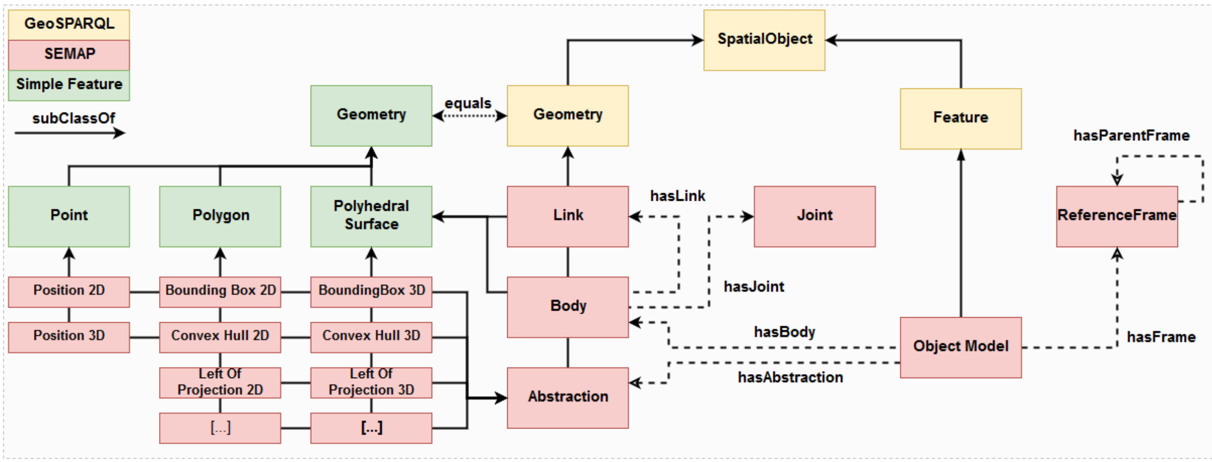


Figure 6.2: An excerpt of the ontology that implements the semantics of SEMAP's environment model.

Figure 6.2 sketches SEMAP's core ontology. It uses standards from the Open Geospatial Consortium (OGC), because these well-defined models of geo-spatial data are in alignment with PostGIS's data types, which were also defined by the OGC. GeoSPARQL's `SpatialObject` and the fundamental distinction between geometries and features are integrated in SEMAP's upper ontology.

Here, the concept `Geometry` describes any kind of spatial primitive and provides a semantic wrapper for all OGC data types and serves as a bridge to the well known Simple Feature Ontology. SEMAP's KB contains a corresponding `Geometry` sub-concept, for every geometric primitives stored in SEMAP's DB. The property `semap:hasDbId` is used to create an associative link between the geometric primitive and its semantic wrapper. SEMAP internally uses these associations to join spatial and semantic data.

The super-concept `Feature` is used for all things that can be described spatially like SEMAP's `ObjectModel`, which aggregates sets of semantically wrapped geometries to represent an object. For this, it uses the `geo:hasGeometry` property and its two specializations: `semap:hasBody` composes a set of geometries that constitute the object's actual body. In case of articulated objects, the `Link` and `Joint` concepts are used to describe the object's kinematics. `semap:hasAbstraction` provides a set of coarser representations, like oriented and axis-aligned bounding boxes and convex hulls. These abstractions are used for accelerated spatial processing and enable the analysis of directional relations like `left-of` or `above-of`, based on projection and half space geometries described in [46].

To create a spatio-semantic environment model for a particular application, domain-specific ontologies, knowledge bases and rule-sets can be imported into SEMAP. To describe domain-specific concepts spatially and reason about them as part of SEMAP's environment model, the respective entities can be associated with an `ObjectModel` via the `semap:hasObjectModel` relation, cf. Figure 6.5 (b).

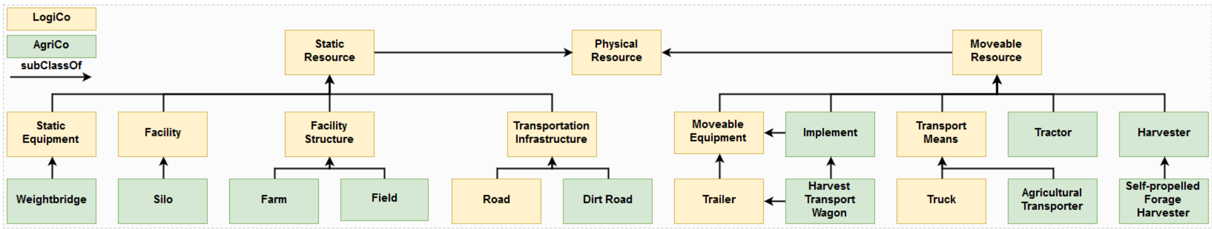


Figure 6.3: Excerpts of the domain-specific model added to SEMAP. The LogiCo ontology (yellow) provides a model of static and movable resources, to which the AgriCo ontology (green) adds agricultural concepts like farms and tractors.

6.4 Applying SEMAP in Agriculture

In this section, we detail the process of customizing SEMAP for a specific application domain. Our goal is to create a spatio-semantic model of agricultural environments and machinery in SEMAP for spatial analysis and rule-based reasoning to derive more information about ongoing agricultural processes that involve multiple machines.

First, we present the description of the semantic model used to represent agricultural concepts, such as fields, farms and tractors in SEMAP's knowledge base. After that we discuss how spatial data is added to this ontological model and how telemetry data recorded from actual agricultural machines can be used to continuously update the constructed environment model. Next, we demonstrate how to use SEMAP for grounding basic spatial predicates between agricultural machines and their environment and how rule-based reasoning is used to identify complex and domain-specific spatial relations. Finally, we present an ontological model for describing agricultural processes in terms of their activities and related events and illustrate how SEMAP's capabilities to answer both spatial and semantic queries, can be used to effectively instantiate the proposed process model to gain more insight into an ongoing agricultural process.

Throughout this discussion, the chosen application example is concerned with the detection of relevant states and events during a maize harvesting campaign, especially the spatial relations between transport vehicle and forage harvester while loading crops.

6.4.1 The AgriCo Ontology

Our semantic model for describing agricultural machinery and their environments is based on the logistics core ontology (LogiCo) by Daniele et al. [73]. This semantic model describes environments and resources in logistics. Since this domain is very similar to the general process of harvesting, we extended LogiCo with additional concepts needed to represent agricultural processes. We call this extended ontology AgriCo as depicted in Figure 6.3.

All components of our model are based on `Physical Resources` in the real world, which can be `Static` or `Movable Resources`. Three sub-classes are used to describe static locations of interest: The `Facility` concept defines areas and structures designated for a specific purpose in the given domain and the `Facility Structure` defines aggregates of different facilities. In AgriCo, for example, `Farm` serves as an aggregate of agricultural facilities like `Silos`. Additionally, the `Static Equipment` concept describes utilities available at a facility, e.g., a `Weightbridge` for weighing transport vehicles. Another important sub-class of static resources are the different kinds of `Transportation Infrastructure` to represent connections between locations. Since this important concept was missing in the LogiCo ontology, we added this concept and suitable sub-classes like `Roads` and `Dirt Roads`.

For movable resources, LogiCo gives concepts for `Transport Means`, i.e., trucks, and `Movable Equipment` such as trailers. Since `Tractors` can not transport goods without an appropriate attachment, AgriCo provides the basic concept `Tractor` as a direct sub-class of movable resource and additionally the `Implement` concept as a specification of movable equipment. It serves as super-concept for the various kinds of machinery that can be connected to a tractor, e.g, plows, sowers. The `hasImplement` relation is used to express that an instance of an implement (or trailer) is attached to an instance of tractor. To describe machinery configurations suitable for agricultural transport activities, AgriCo defines the generic `Agricultural Transporter` concept. Instances of this concept need to be asserted or derived, i.e., by the rule shown in Figure 6.4.

Based on these concepts it also possible describe very specific agricultural resources such as `Harvest Transport Wagons`, which inherit properties from the trailer and implement concept simultaneously. In this way, we can denote the trailer's volumetric capacity via the `logico:hasCapacity` attribute, as well as the interfaces use to control the active pickup systems and scraper floor via the `agrico:hasISOBUSInterface` relation.

Finally, the `Harvester` concept is used to represent combine and forage harvesters, which are directly derived from the `Movable Resource` concept, too, as they can not be used for transporting goods in a supply chain.

```
?tractor rdf:type agrico:Tractor
?trailer rdf:type logico:Trailer
?tractor agrico:hasImplement ?trailer
==>
?tractor rdf:type agrico:AgriculturalTransporter
```

Figure 6.4: A rule for inferring an agricultural transport vehicle from its connected parts.

6.4.2 Instantiating the Environment and Machinery Model

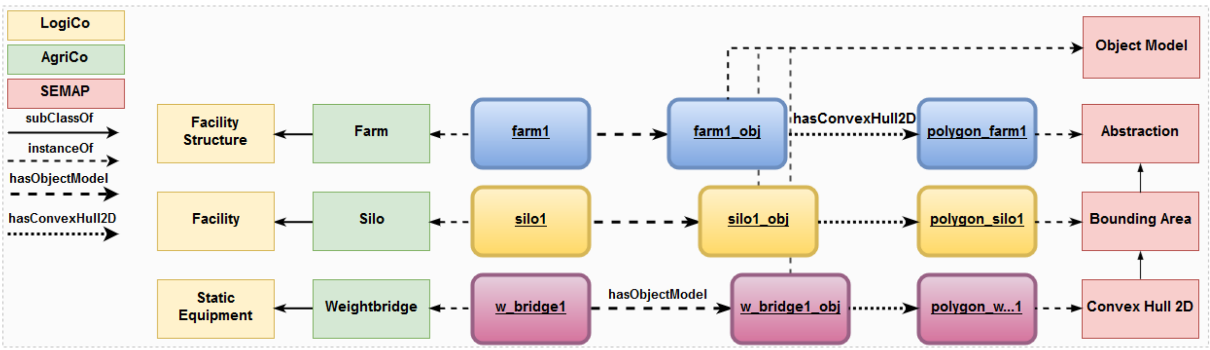
The semantic model presented so far provides the conceptual basis from which instances of agricultural facilities and machinery can be created and described. To link them to a spatio-semantic data set in SEMAP, we proceeded as follows:

First, we imported the AgriCo ontology into SEMAP's KB component. Next, we allowed that the `hasObjectModel` property can map from instances of LogiCo's `Physical Resource` to SEMAP's `ObjectModels`. This way, the domain-specific concepts and instances thereof can have a spatial representation in SEMAP. Finally, we instantiated the agricultural concepts and their spatio-semantic representation with an appropriate data set.

To setup static resources in our environment model, we used a set of polygonal boundaries to represent farms and fields and other facilities. Figure 6.5 (a) shows an excerpt of the environment. It consists of the farm's grounds (blue), three silos (orange) and a vehicle scale (violet), as well as two fields (green). The data was modeled in Google Earth and automatically read into

SEMAP's KB and DB components using a KML file importer. In Figure 6.5 (b), the underlying semantic representation is depicted with three instances of AgriCo concepts related to their object representation using the `hasObjectModel` relation. Here `farm1` connects to `farm1_obj`. The polygonal boundary `farm1_boundary` is connected via the `hasConvexHull2D` property, which is a sub-property of `hasAbstraction`.

(a) The spatial data used to represent a farm (incl. silos) and two fields.



(b) The semantic representation within SEMAP's knowledge base.

Figure 6.5: To represent a farm's facilities in SEMAP, we used the 2D polygonal boundaries, shown in (a), stored in the DB component. These spatial models are connected to instances of the domain-specific concepts of AgriCo via SEMAP's `ObjectModel` concept, as illustrated in (b).

To add movable resources to the static environment, we created three-dimensional and articulated object models of a tractor-trailer combination and a forage harvester as displayed in Figure 6.7 (b). These objects are modeled in the Unified Robot Description Format, since SEMAP supports this format natively. The underlying semantic representation is a straight-forward extension to the example in Figure 6.5 (b), yet more complex due to the individual links and joints.

To introduce movement to our spatio-semantic model of farms and fields, we used telemetry data recorded on real agricultural machines to continuously update the position and articulation of the machines within it. We replayed the machine's GPS signals and joint states in the Robot Operating System (ROS) and connected a bridge node to SEMAP, such that the environment model was updated accordingly.

6.4.3 Analyzing Spatial Relations

By moving the agricultural machines through the static environment in our experimental setup, the spatial relations between environment and machines and the machines themselves are changed continuously. SEMAP's spatial and semantic reasoning capabilities can be used to detect these spatial relations using geometric analysis and express them in terms of semantic spatial predicates.

SEMAP provides spatial operators to test for containment and intersection in 2D and 3D, as well as operators to identify directional relations, i.e., left-of, right-of. The same holds for distance-based relations, such as near-by or far-away, which can be parameterized to set a desired distance threshold. For a full discussion on SEMAP's spatial operators, see [3].

To use theses operators for reasoning about spatial relations between machines and their environment, we follow a two step procedure:

First, we make use of SEMAP's qualification capabilities to geometrically ground the spatial relation of interest by posing a suitable query to SEMAP's DB backend via its ROS interface. Figure 6.6 (a) gives an example on how to test for containment between pairs of SEMAP's object models. The query identifies objects of type `Facility` as reference and those of type `MovableResource` as the targets. The query is further parameterized to uses the movable resource's 2D position for the geometric evaluation against the 2D convex hull of the facilities. Hence the given query performs quantitative spatial analysis, between the agricultural machines in our model and the surrounding environment by checking whether a machine's 2D position is spatially in a facility's boundary. The results of this query are then inserted into SEMAP's knowledge base as qualitative semantic knowledge about the spatial relations. In case of our example, the objects pairs found by the query are inserted as facts over the `semap:isIn2D` relation. Likewise SEMAP's ontological model defines relations such as `semap:leftOf2D` or `semap:containedIn3D`, which are extracted by the same query process.

Secondly, we use the derived spatial knowledge in order to reason about our agricultural application domain. For example, we can infer that for all pairs of machinery and environment entities for that the spatial predicate `semap:isIn2D` holds, the topological relation `logico:isAt` – which is defined in our domain ontology – holds, too. An example for such rule-based inference is given in Figure 6.6 (b). Here, the rule identifies to topological location of a movable resource (i.e., a tractor) based on the spatial relation to any of the facilities contained in our model (i.e., fields and farms).

While this seems a simple transition, it is important to note that this rule infers from a *spatial* predicate to a *topological* relation and that this assertion is grounded in the quantitative *geometric* data within SEMAP's DB. The rule is generic for all instances of `Movable Resource` at any instance of `Facility` and its sub-concepts, which makes it applicable in a wide range of applications. The underlying spatial querying is done automatically in SEMAP's multi-modal query interfaces, such that further queries to the environment model can be posed using the high-level relation `isAt`, without having to deal with the data transfer from DB to KB explicitly.

It is also important to note that the transition from spatial to topological information is explicitly coded through the shown inference rule. It is thus a matter of application design, how to implement this transition. Instead of using 2D containment, we could have also opted for grounding topological relations using a 3D spatial containment relationship or work with distance-based constraints.

```
rosservice call /containment_query
"reference_object_types: ['Facility'] reference_object_geometry_type: 'ConvexHull2D'
target_object_types: ['MovableResource'] target_object_geometry_type: 'Position2D'
fully_within: false insert_kb: true"
```

(a) SEMAP query to extract containment relations.

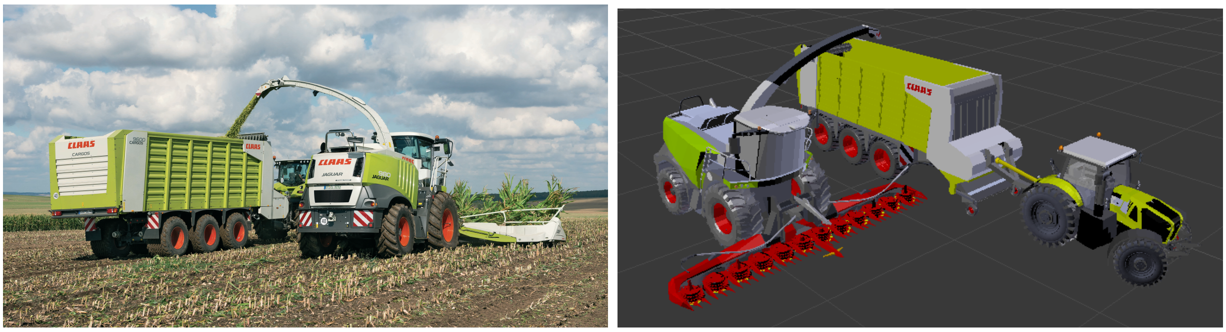
```
?machine rdf:type logico:MovableResource
?machine semap:hasObjectModel ?machine_obj
?machine_obj semap:hasPosition2D ?machine_abstr_pos2D
?facility rdf:type logico:Facility
?facility semap:hasObjectModel ?facility_obj
?facility_obj semap:hasConvexHull2D ?facility_abstr_ch2D
?machine_abstr_pos2D semap:isIn2D ?facility_abstr_ch2D
==>
?machine logico:isAt ?facility
```

(b) Rule to ground topological relations based on spatial relations.

Figure 6.6: To geometrically ground spatial containment relations, we used the query shown in (a). The query results where extracted into SEMAP's KB as facts over the `semap:isIn2D` relation and then used the rule (b) to derive that the topological relation `logico:isAt` holds between machines and facilities.

It is this flexible approach in spatio-semantic reasoning that makes SEMAP beneficial when extracting information about a given application domain. To refine, for example, the generic `logico:isAt` relation to provide more information about our agricultural scenario, we extended AgriCo to provide additional sub-relations for the most important facility types in our model, such as `agrico:onFarm` and `agrico:onField`, which are extracted as explicit semantic facts via an additional set of rules.

Similarly, we can use the same type of reasoning to analyze spatial relations between a pairs of machines. For example, we used SEMAP to detect that a transport vehicle (TV) is correctly positioned for a loading procedure, due to its directional relations regarding a self-propelled forage harvester (SFH). Figure 6.7 exemplifies how to construct this complex domain-specific relation by combining several basic spatial relations with additional domain-dependent knowledge. Figure 6.7 (a) depicts the situation of interest in real life, whereas (b) shows visualization of a similar scene represented in SEMAP. To identify that the transport vehicle is properly positioned for loading, the rule shown in (c) checks the trailer's 2D convex hull for containment in the harvester's `left-of` projection, to verify that the transport vehicle is left-of the harvester. If so, the relation `agrico:positionedForLoading` is inferred to hold between the transport vehicle and the harvester.



(a) Loading in reality. **(b)** Loading in RViz.

```
?sfh rdf:type agrico:Harvester
?sfh semap:hasObjectModel ?sfh_obj
?sfh_obj semap:hasLeftOfProjection2D ?sfh_proj_l2D
?tv rdf:type agrico:TransportVehicle
?tv semap:hasObjectModel ?tv_obj
?tv semap:hasConvexHull2D ?tv_abstr_ch2D
?tv_abstr_ch2D semap:isIn2D ?sfh_proj_l2D
==>
?tv agrico:positionedForLoading ?sfh
```

(c) The rule for grounding the `positionedForLoading` relation in SEMAP.

Figure 6.7: We used telemetry data from an actual loading procedure (a), to move and articulate the machines in ROS and visualize them in RViz (b). We also synchronized the telemetry with our SEMAP model and used the rule (c) to identify the correct spatial positioning of two machines for loading harvested goods from a forage harvester onto a transport vehicle.

This kind of reasoning deduces a valuable symbolic representation about the underlying agricultural process, which was previously covert in the telemetry data of both machines. Here, SEMAP's spatio-semantic processing makes this information explicitly available as factual knowledge within SEMAP's KB. Such a representation is useful for further processing, for example, to monitor changes of the spatial relations over time. Especially when looking at logistic problems in harvesting processes, the spatial transitions of resources correspond strongly with the underlying process the machines go through. For example, a transporter arriving at the harvester initiates loading or being on a silo corresponds to unloading a trailer. To account for such situations, we extended SEMAP's core ontology further to support reasoning about process states in such contexts. We are currently not aware of similar works in the literature.

6.4.4 The AgriServ Ontology

Next, we describe the AgriServ ontology extension that allows to describe agricultural work and services in terms of the activities that have to be performed to achieve a certain logistical objective in the agricultural domain. It is again based on work by Daniele et al. [73] and also relies on ideas proposed by Hoxha et al. in [114]. Figure 6.8 shows an excerpt of the ontology.

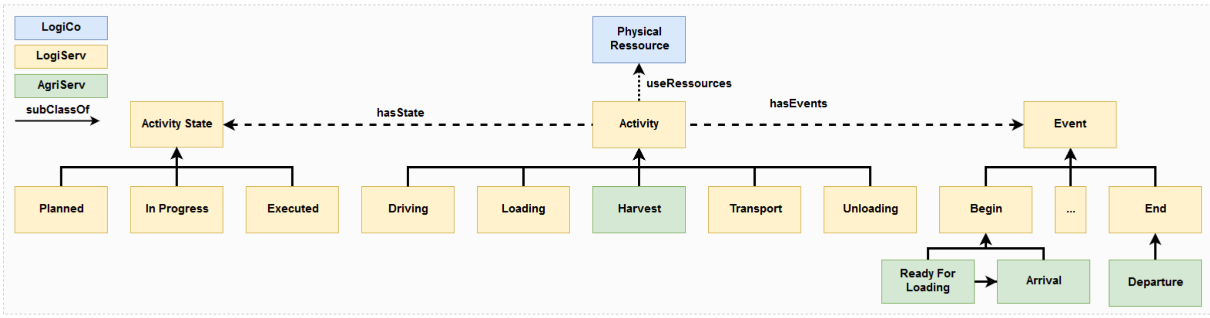


Figure 6.8: An excerpt of the AgriServ ontology that provides a model of agricultural processes.

The description of agricultural processes in AgriServ revolves around the concepts of activities and events. The `Activity` concept describes the actionable steps of an logistic transport process, i.e., loading goods at a origin location A, transporting them from A to B and unloading them at their destination location B. There are also activities defined that are specific to the agricultural domain, such as harvesting crops. To denote which resources are involved in an activity, the `logiserv:usesResource` relation is used. It maps from the instance of an activity concept to one or many instances of the `PhysicalResource` concept defined in the upper-ontologies LogiCo and AgriCo. This relation can, of course, be further differentiated to specify the requirements towards a certain type of activity. For example, AgriServ defines the relations `hasField`, `hasHarvester` and `hasTransporter` to clarify on the specific roles of the `Loading` activity. The spatial locations at which the activity begins and ends are denoted through the relations `hasOrigin` and `hasDestination` which point to an instance of `StaticResource`, pointing to one of the agricultural facilities introduced in AgriCo. Likewise, each activity can be annotated with the time frame in which it is valid, using the `hasBegin` and `hasEnd` relation to point to a specific time stamp. The semantics of this time interval may vary due to the status of the given activity. An activity's state is reflected through the `hasState` predicate which points to an element of a fixed set of progress states, namely `Requested`, `Planned`, `In Progress` or `Executed`.

Closely related to the state of an activity are the events associated with it. Each `Event` denotes a significant occurrence during the activity's life cycle and maybe the cause of changing an activity instance's current state. To differentiate between different types of events, AgriServ uses sub-classes. It provides basic event types, such as `Begin`, `End`, `Suspend` and `Resume`, to describe the general progress of an activity. Each factual instance of event identifies a resource as its subject, as well as another resource as its target, if this applicable, like in cooperative activities such as loading crops from a harvester onto a transport vehicle. An event also gives a time stamp and location, denoting when and where it occurs, too.

Activities are described as a sequence of events and hold a list of associated instances via the `hasEvent` relation. This relation is further differentiated by sub-relations, which carry a specific semantic relative to an activity's state. The `hasPlan` relation, for example, maps to all the expected events of a planned activity, whereas the `hasTrigger` relation identifies all events that progress the activity regardless of whether the plan is matched or not. Finally, the `hasActual` relation maps to all events that actually occurred during an activities execution.

The spatial state transitions of a movable resources within its environment are nothing short of events, it is therefore useful to provide concepts for spatial events, too. AgriServ provides the events `Arrival`, `Departure`, as spatially related refinements of the begin and end events, which always need an additional resource assignment to identify the target it is in reference to. Similarly, the `ReadyForLoading` event is issued based on the domain-specific spatial relation `positionedForLoading`.

6.4.5 Mapping from Spatial Events to Process Events

To inspect the changing spatial relations in our application example, we queried SEMAP for the relevant relations with every incoming telemetry datum. In our experiments, we sampled telemetry data at a rate of 1 hz to generate a continuous trace log of the machines' whereabouts and their relations towards each other. This sub sampling was done to reduce the amount of collected data to a reasonable size while keeping enough temporal resolution to trace and detect relevant events.

Figure 6.9 gives an example of such a trace. It shows how `tractor1` arrives at the farm, visits the vehicle scale and then continues to drive to the silo, as it goes through the process of weighing its load and then unloading it at the silo. The trace shows further that at the same time `harvester1` is arriving at `field2`, where it is approached by `tractor2` shortly after. This approach can be monitored through different stages, as `tractor1` first comes near the harvester indicated by the `inDistance` relation and then takes the correct position for loading, as discussed above. In both cases, the spatial transitions give strong indications about the underlying agricultural process, hence we went on creating spatial events and mapped them onto the process model.

```
# Time     | # Reference  | # Spatial Relation     | # Event
 | # Target
13:16:45 | tractor1     | onFarm                 | Arrival   | farm1
13:16:46 | tractor1     | onVehicleScale         | Arrival   | scale
13:16:51 | harvester1   | onField                | Arrival   | field2
13:16:53 | tractor1     | onVehicleScale         | Departure | scale
13:16:59 | tractor1     | onSilo                 | Arrival   | silo_north
13:17:02 | tractor2     | onField                | Arrival   | field2
13:17:36 | tractor2     | inDistance             | Arrival   | harvester1
13:17:45 | tractor2     | positionedForLoading   | Arrival   | harvester1
13:18:03 | tractor1     | onSilo                 | Departure | silo_north
13:18:35 | tractor1     | onFarm                 | Departure | farm1
13:20:21 | tractor2     | positionedForLoading   | Departure | harvester1
13:20:29 | tractor2     | inDistance             | Departure | harvester1
13:21:28 | tractor2     | onField                | Departure | field2
```

Figure 6.9: A continuous trace log of spatial relations between machines and environment created through analyzing telemetry data with SEMAP.

Since SEMAP's query system is stateless and processes each query on the current world state of the environment model independently, there is no immediate tracking of previous states. Event generation is currently done in an external processing node which accounts for the state history by comparing timestamps and generates the appropriate events, if a spatial transition occurs. When, for example, the fact `tractor1 isAt farm1` did hold at timestamp t_n, but does no longer hold at t_{n+1}, an `Departure` event is created and asserted to the KB, cf. Figure 6.9.

In the same way, we approached the detection of process states and events. To trace the harvesting process, an additional processing node was setup to encodes a state machine that inspects the spatial events using simple transitioning rules and creates the process events accordingly. Here we exploit the fact that the process states occurring during the harvesting operation are in fact spatially disjunct. Figure 6.10 shows the mapping onto the process events for the same dataset that was used in Figure 6.9. It shows how spatial arrival at the farm triggers the beginning of the process state `Farmwork` and how the spatial relation `positionedForLoading` is used to ground the start of a loading procedure.

```
# Time     | # Reference  | # Process
| # Event  | # Target
13:16:45 | tractor1     | Farmwork   | Begin  | farm1
13:16:46 | tractor1     | Weighing   | Begin  | scale
13:16:51 | harvester1   | Fieldwork  | Begin  | field2
13:16:53 | tractor1     | Weighing   | End    | scale
13:16:59 | tractor1     | Unloading  | Begin  | silo_north
13:17:45 | tractor2     | Loading    | Begin  | harvester1
13:18:03 | tractor1     | Unloading  | End    | silo_north
13:18:35 | tractor1     | Farmwork   | End    | farm1
13:20:21 | tractor2     | Loading    | End    | harvester1
```

Figure 6.10: A continuous trace log of process events created through analyzing the spatial transitions using a state machine.

6.4.6 Reasoning about Activity Sequences

The above example demonstrates, how to ground single process events as instances that hold at a given point in time. This is the first step towards constructing instances of process states facts that hold true for a certain time interval. Such states can be constructed by determining pairs of associated `Begin` and `End` instances of a fixed event type occurring in the event sequence of a particular agricultural machine. When the respective interval is closed, this is detected by the reasoning node and an additional process event is issued. Then the relation `hasState` is instantiated as `Executed` and the event's type is declared as `Interval`. Afterwards, all the intermediary events are associated with the constructed interval for proper reference.

The construction of interval-based process states allows to reason about process durations and other key performance indicators. Having access to the harvester's telemetry, for example, enables us to link a current measurement of the machine's total yield counter to the `Begin` and `End` events of a particular `Loading` sequence. When constructing the respective process interval, it is then possible to provide an estimate of the total mass loaded onto the transport vehicle during the process sequence, simply by subtracting the two measurements associated with the event instances. This is highly valuable information is automatically inferred through our system and its spatio-semantic reasoning capabilities and gives the analyst monitoring the harvesting campaign directly valuable information about the performance of the involved machinery.

Obviously, the same style of reasoning can be extended towards reasoning about more complex activity patterns. For example, we can also detect more abstract, not immediately spatially related process states of transport vehicles by applying template matching over process intervals. A rule like shown in Figure 6.11 can be used to identify a sequence of executed process states, which correspond to a full transport cycle between field and farm, that is a chain of intervals `Loading`, `Transport`, `Unloading` and `Driving` in immediate temporal succession.

Once such a sequence is detected, an additional process state of the class `Transport Cycle` is created. Applying this reasoning over time, we can amass several instances of this high-level process state, as the transport vehicle completes several transport runs from field to farm and back. This sequence detection is another valuable feature of our approach, as it can be used to further aggregate process information and present it in a semantic format.

```
?tv rdf:type agrico:TransportVehicle

?tv logiserv:performsActivity ?loading

?loading rdf:type agriserv:Loading
?loading logiserv:hasType loading_type
?loading_type rdf:type logiserv:Interval
?loading logiserv:hasState ?loading_state
?loading_state rdf:type logiserv:Executed

?tv agriserv:performsActivity ?transport

?transport rdf:type agriserv:Transport
?transport logiserv:hasType transport_type
?transport_type rdf:type logiserv:Interval
?transport logiserv:hasState ?transport_state
?transport_state rdf:type logiserv:Executed

?transport temporal:immediatelyAfter ?loading

?tv logiserv:performsActivity ?unloading

?unloading rdf:type agriserv:Unloading
?unloading logiserv:hasType unloading_type
?unloading_type rdf:type logiserv:Interval
?unloading logiserv:hasState ?unloading_state
?unloading_state rdf:type logiserv:Executed

?unloading temporal:immediatelyAfter ?transport

?tv logiserv:performsActivity ?driving

?driving rdf:type agriserv:Driving
?driving logiserv:hasType driving_type
?driving_type rdf:type logiserv:Interval
?driving logiserv:hasState ?driving_state
?driving_state rdf:type logiserv:Executed

?driving temporal:immediatelyAfter ?unloading
==>
?transport_cycle rdf:type agriserv:TransportCycle

?cycle_type rdf:type logiserv:Interval
?transport_cycle logiserv:hasType ?cycle_type
?cycle_state rdf:type logiserv:Executed
?transport_cycle logiserv:hasState ?cycle_state

?tv logiserv:performsActivity transport_cycle
```

Figure 6.11: Identification of a completed transport cycle as a sequence of individual process state intervals, by inspecting their temporal relations.

6.5 Conclusion and Future Work

In this article, we used the SEMAP framework for combined spatial and semantic reasoning about machine-environment and machine-machine in an agricultural domain. We showed that the core concepts of SEMAP presented in [3] scale well into other application while re-using the its core ontology as an upper ontology for more domain specific models. To our knowledge, there is currently no comparable framework described in the literature that combines spatial reasoning, topological relations and semantic background knowledge in such a flexible way.

Besides providing a proof-of-concept of SEMAP's capabilities, we created two ontological models for process modeling in agricultural domains. We extended an ontological model from the logistics domain resulting in the agricultural core ontology AgriCo to model agricultural resources and their relations. Additionally, we developed the AgriServ ontology to describe agricultural work and services in terms of activities and related events. Based on this semantic model, we instantiated a data set that combined factual knowledge with spatial data in our framework. Using recorded telemetry data, we moved and articulated several agricultural machines to replay a forage maize harvesting campaign. We used SEMAP's spatial operators for quantitative spatial analysis to classify topological relations between fields and machines. We also used an ontological model of logistical and agricultural processes and rule-based reasoning over the changing relations, to detect process states and events relevant to the harvesting process. We exemplified this process by showing how to infer that a transport vehicle is ready for loading due to its position relative to the harvester. Based on such relations, we were able to detect ongoing processes and relevant events, namely the begin and end of different high level processes.

Our approach demonstrates that the use of semantic mapping technology in agriculture is beneficial, as we were able to extract valuable information about the agricultural process out of the geo-referenced stream of telemetry data. The derived knowledge about machine-machine and machine-environment relations is validated in the geometric state of the environment and also available as machine-readable facts that adhere to a formal ontological model, which opens up possibilities for the further development of decision support systems.

Currently, the data model is updated continuously to represent the environment's current state, but provides neither a history of past states, nor methods to query about temporal change. This denies the possibility to detect events by querying the temporal sequence of certain relations and states directly. For this, we relied on additional external processing modules coupled with SEMAP to detect events. Adding a temporal information layer to SEMAP will be a necessary next step to realize proper temporal analysis and event generation. Such an extension of the SEMAP system towards proper temporal reasoning based on an ontological model and rule-based reasoning is straight forward. Approaches like the ones presented in [22, 102] will be included in future work. Additionally, stream reasoning approaches like the Continuous SPARQL framework (CSPARQL) [18] could be used. Efficient stream reasoning would also allow to include additional more telemetry data then just geo-location. The inclusion of such additional data will allow quantitative assessment of the harvesting process, which would complement our current approach of qualitative evaluation.

An open problem is the deployment of the system in the context of a real-world agricultural application. While we used telemetry data recorded in an actual harvesting operation, we conducted our experiments in the lab without the implications of actual field operations. To

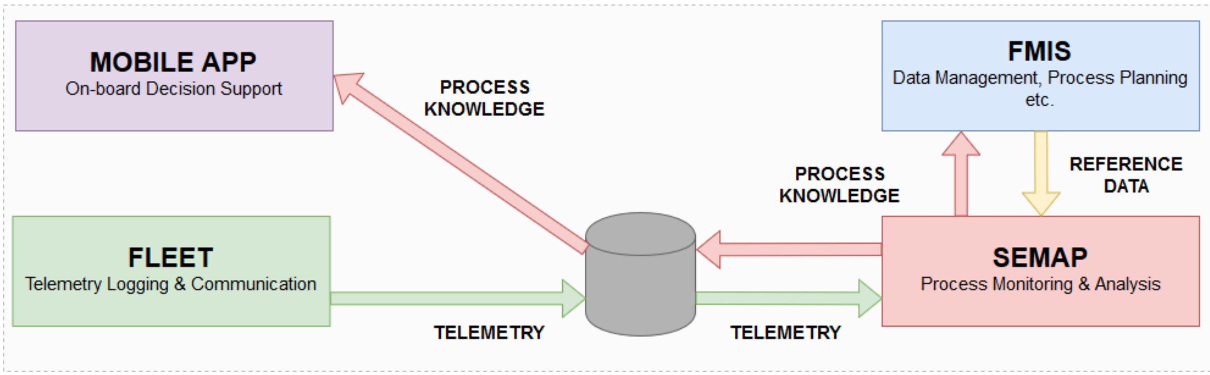


Figure 6.12: In the context of a decision support architecture for agricultural processes, SEMAP could receive telemetry data from a fleet of agricultural machines and process it in real time to provide knowledge about the on-going agricultural operation. The information could be used to assist the machine operators via on-board mobile apps; or the operation's manager, ie. a farmer, via a Farm Management Information Systems (FMIS).

proceed in this direction, we intend to use the SEMAP system in the context of a decision support architecture for agricultural processes. Figure 6.12 shows a sketch of such an architecture. To provide decision support for machine operators and process managers, our system could be coupled with a fleet of agricultural machines sending telemetry data during operations. This data would then be processed using SEMAP to generate process knowledge, which could be used to assist the machine operators via on-board assistance systems implemented as mobile apps. Likewise, SEMAP could be coupled with Farm Management Information Systems (FMIS) [121] to receive relevant reference data, such as environment information or pre-planned tasks and provide process knowledge for subsequent processing by the farmer or additional tools.

Working towards such an architecture would require to provide the presented process monitoring capabilities in real time, which is currently not feasible with the used techniques. Such a distributed system would also have to robustly cope with situations where the data stream is disrupted by insufficient connectivity with the fleet of agricultural machines. As agricultural environments are often poorly covered with mobile, suitable coping mechanisms must be found, e.g., by predicting future states. Here incorporating reasoning under uncertainty may assist. Exploring these topics is also subject to future work.

Chapter 7

Smart Contracts and Smart Payment for Farming 4.0

Dominik Sparer Fraunhofer-Institut für Materialfluss und Logistik IML, Dortmund, Germany
Henning Deeken CLAAS E-Systems KGaA mbH & Co KG, Dissen, Germany
Björn Künsting Diebold Nixdorf Inc., Paderborn, Germany
Philipp Sprenger Fraunhofer-Institut für Materialfluss und Logistik IML, Dortmund, Germany

Zusammenfassung

In diesem Beitrag wird dargelegt, wie Digitalisierung von Geschäftsprozessen am Beispiel der Zusammenarbeit zwischen Landwirten, Lohnunternehmen, Betreibern von Biogasanlagen und Finanzdienstleistern in der Landwirtschaft erfolgen kann. Hierbei sind die sichere Vernetzung, die Datentransparenz und die nachvollziehbare Speicherung von Prozessänderungen von zentraler Bedeutung. Ein wichtiger Ansatz für die Digitalisierung der unternehmensübergreifenden Zusammenarbeit ist die Weiterentwicklung und Anwendung der Blockchain-Technologie für den betrachteten landwirtschaftlichen Anwendungsfall. Aufbauend auf der Blockchain-Technologie wird in diesem Beitrag ein Ansatz zur Automatisierung mithilfe von Smart Contracts und Smart Objects vorgestellt. Neben dem Tracking des Ernte- und Transportprozesses liegt der Fokus insbesondere auf der Anbindung einer Payment-Plattform an die Blockchain und dem entsprechenden Regelwerk, welches mittels Smart Contracts eine automatisierte und papierlose Transaktion ermöglicht. Zur Prozessbeschleunigung und zur automatischen Prozessüberwachung werden die Landmaschinen in der logistischen Kette mit dezentralen Steuereinheiten (Smart Objects) ausgerüstet.

7.1 Einleitung

Im Kontext von Industrie 4.0 wird eine digitale Transformation mit immer weiterer Vernetzung von Ressourcen in Arbeits- und Produktionsprozessen angestrebt. Durch den Einsatz von cyberphysischen Systemen (eng. Cyber-Physical Systems (CPSs), siehe Abschnitt 7.2.3) ist es heute möglich, relevante Daten auf Prozessebene zu erheben und an zentrale ERP Systeme zu senden, welche anschließend aussagekräftige Kennzahlen aggregieren und verarbeiten [124]. Diese Art der Digitalisierung von Geschäftsprozessen hat sich schon in vielen Industriezweigen durchgesetzt und umfasst die eingesetzten Arbeitsmaschinen und Produkte sowie auch den Menschen. Oftmals steht hierbei eine optimierende Planung und Steuerung der Arbeitsprozesse im Vordergrund.

Die Digitalisierung von Dienstleistungen schließt auch die digitale Abbildung und automatische Bearbeitung von Geschäftsprozessen ein. Insbesondere wenn automatisierte Arbeitsprozesse mit digitalisierten Geschäftsprozessen verknüpft werden, ergeben sich große Potenziale, da Leistungen schon während ihrer Erbringung erfasst und bewertet werden können. So lassen sich beispielsweise Abweichungen von digital festgeschriebenen Rahmenbedingungen frühzeitig aufdecken und korrigieren oder aber der erfolgreiche Abschluss einer Dienstleistung prüfen, um Folgeaktivitäten, zum Beispiel die Abrechnung, zu automatisieren [62].

Um diese Potenziale auszuschöpfen, muss eine enge Verknüpfung der agierenden CPS auf der Arbeitsebene mit den digitalen Verträgen auf der Geschäftsebene realisiert werden. Ebenfalls müssen externe Systeme, die für die Abwicklung einer Dienstleistung relevant sind (z. B. Finanzdienstleister, ERP-Systeme), eingebunden werden [161].

Hierbei ist es wichtig, die Vertraulichkeit sensibler Geschäftsdaten zu gewährleisten und zugleich Transparenz über die ablaufenden Prozesse für die beteiligten Partner und technischen Systeme zu schaffen. Besonders private Blockchains mit einem eingeschränkten, bekannten Nutzerkreis sowie geregelten Lese- und Schreibrechten können im B2B-Kontext die passende Balance zwischen Privatsphäre, Fälschungssicherheit und Transparenz bieten [62].

Die aufkommende Blockchain-Technologie ermöglicht es, über digitale Vertragswerke (sogenannte Smart Contracts) eine autonome dezentrale Entscheidungsfindung zu schaffen. Sowohl Smart Contracts als auch private Blockchains sind aber noch sehr unerforscht. Mit Ausnahme von Kryptowährungen sind bisher nur wenige konkrete Anwendungen für Blockchains vorgestellt worden.

Daher soll an dieser Stelle die Anwendung als praxisnahes Beispiel für den Einsatz privater Blockchains vorgestellt werden. Der Artikel demonstriert, wie die Koppelung mehrerer Blockchains eine effektive Schnittstelle zwischen digitalen Verträgen, einer digitalen Prozessnachverfolgung auf Basis von CPSs und einer Plattform für automatisierte Bezahl- und Rechnungsvorgänge realisieren kann.

Das entwickelte System ist in der Lage, die Geschäftsprozesse rund um eine logistische Kette abzubilden, die Leistungserbringung automatisiert nachzuverfolgen und vorab bestimmte Schritte durch eine implementierte Business-Logik automatisiert abzuwickeln. Im Folgenden wird dies an den Geschäftsbeziehungen zwischen Landwirten, einem Lohnunternehmer und einer gewerblichen Biogasanlage im Kontext einer landwirtschaftlichen Silomaisernte verdeutlicht.

Die vorgestellten Ergebnisse wurden im Rahmen des vom BMBF geförderten Projekts *Prozessinnovation in Planung und Steuerung von Wertschöpfungsnetzwerken durch Integration von Smart Objects und Smart Finance Ansätzen (SOFiA)* [169] erarbeitet. Dieses Projekt hat die Automatisierung und Dezentralisierung von Planungs und Steuerungsprozessen in logistischen Netzwerken sowie der zugrundeliegenden Geschäftsprozesse und finanztechnischen Transaktionen zum Ziel. Entwickelt wird das SOFiA-System im Kontext von zwei Anwendungsfällen, einerseits dem Transport von Containern über eine klassische, multi-modale Supply-Chain sowie andererseits der Ernte von Silomais, einem logistisch aufwendigen Agrarprozess.

Im Projekt arbeiten das Fraunhofer Institut für Materialfluss und Logistik (IML), der

Landmaschinenhersteller CLAAS, der Informationstechnikkonzern Diebold Nixdorf und der Logistikdienstleister EKOL zusammen.

Für allgemeine Informationen zum Projekt SOFiA, siehe [1] und [225, 226].

7.2 Grundlagen

7.2.1 Blockchain

Die Blockchain ist eine neue, sichere, verteilte und insbesondere fehlerresistente Technologie für internetbasierte Transaktionen [184].

Sie hat das Potenzial, ohne den Einsatz von Intermediären Vertrauen zwischen den vielfältigen Beteiligten der globalen Wertschöpfung herzustellen. Sie ermöglicht Transaktionen monetärer Werte, den Transfer von Daten sowie die Abwicklung von Verträgen und stellt dabei Nachvollziehbarkeit im Sinne von Prüfungen und Revisionen sicher [184].

Grundsätzlich wird bei auf dem Markt befindlichen Blockchain-Lösungen zwischen Public und Private Blockchains unterschieden. Die Public Blockchains werden vorrangig für das Handeln und Tauschen von Kryptowährungen genutzt. Das bekannteste und gleichzeitig älteste Beispiel aus dieser Gruppe ist die Bitcoin-Blockchain. Sie beruht auf dem 2008 unter dem Synonym „Satoshi Nakamoto“ veröffentlichen Whitepaper [149]. Alle Blockchain-Lösungen die sich nur mit dem Austausch von Kryptowährungen beschäftigen, werden in die Gruppe „BlockChain 1.0“ eingeteilt [204].

Der Verbindungsaufbau zu einem Public-Blockchain-Netzwerk ist nicht reglementiert, wodurch die Daten ohne Autorisierung geladen und auch eingesehen werden können. Auf den Einsatz einer Verschlüsselung der Daten wird bewusst verzichtet, um die Konsensmechanismen der Blockchain nicht zu beeinträchtigen. Darüber hinaus verfügt jeder Teilnehmer über die gleichen Rechte und kann neue Blöcke einfügen, wenn diese dem Konsensmechanismus entsprechen und vom Netzwerk bewilligt werden. Eine Authentifizierung und der Einsatz von Kryptografie ist beim Schreiben von Blöcken auch in einer Public Blockchain notwendig, um die Identität der Teilnehmer zu bestätigen und die Transaktion zu signieren [149].

Mit der Ethereum-Blockchain wurde zum ersten Mal ein dezentrales Turing-vollständiges System eingeführt, auf dem Smart Contracts ausgeführt werden können [127, 217]. Die Ethereum-Blockchain gehört zu den Public Blockchains und setzt ebenfalls eine eigene Kryptowährung für ihre Smart Contracts ein. Die Einführung von Smart Contracts war der nächste große Schritt, der zur Entwicklung der Blockchain-Technologie beitrug. Durch den großen Einfluss auf die weitere Entwicklung werden alle Blockchain-Lösungen, die den Ansatz des Smart Contracting verfolgen, in die Gruppe der „Blockchain 2.0“ eingeteilt [204].

Die größten Kritikpunkte an Public-Blockchain-Lösungen sind der hohe Stromverbrauch und die hohen Latenzen, die durch das Mining und für das Anfügen neuer Blöcke anfallen. Durch den Konsensmechanismus, der durch das gesamte Netzwerk ausgeführt werden muss, steigt der Rechenaufwand für das Verifizieren und Schreiben eines neuen Eintrags immens, desto größer die Blockchain wird und umso mehr Server am Netzwerk teilnehmen [220].

Die Private Blockchain wurde für den Einsatz in einem abgeschlossenen Konsortium entwickelt, um Daten sicher, einfach und transparent zu speichern. Für den Verbindungsaufbau zu einem

Private-Blockchain-Netzwerk muss sowohl eine Authentifizierung als auch eine Autorisierung erfolgen, um die Vertraulichkeit der Daten zu gewährleisten. Durch die Zugriffsbeschränkung kommt die Private Blockchain häufig im industriellen Umfeld zum Einsatz. Die Berechtigung, einen neuen Server in das Netzwerk zu integrieren, wird vom bestehenden Blockchain-Konsortium gemeinsam erteilt. Um zusätzlich zum Recht eine Verbindung aufzubauen, können bei Private Blockchains meist Lese-, Schreib- und Administrationsrechte über ein Berechtigungsmanagement einzeln vergeben werden [184].

Die neueste Generation der Blockchain-Lösungen wird unter dem Begriff „Blockchain 3.0“ zusammengefasst. Sie beschäftigt sich mit Anwendungen, die keinen direkten Bezug zu Finanzen oder Währungen haben. Ein Hauptaugenmerk liegt dabei auf der Skalierbarkeit und der Verarbeitung von Transaktionen [204]. Im Hinblick auf das Internet of Things (IoT) beschäftigen sich Blockchain-Entwicklungen wie „IOTA“ mit der Verarbeitung von großen Mengen an Transaktionen innerhalb kürzester Zeit.

Gegenstand dieses Beitrags ist der industrielle Einsatz der Blockchain-Technologie in einem landwirtschaftlichen Anwendungsfall mit bekannten Geschäftspartnern und somit werden im Folgenden nur Private Blockchains näher betrachtet.

Weitere Grundlagen zur Blockchain-Technologie können dem Blockchain-Positionspapier der Fraunhofer-Gesellschaft entnommen werden [184].

Verifizierung und Verteilung neuer Blöcke Der einmal eingetragene Block kann nicht mehr verändert werden. Das ist ein unumstößliches Paradigma der Blockchain-Technologie. Daher ist es umso wichtiger, dass Blöcke validiert sind, bevor diese geschrieben werden [184].

Jeder Teilnehmer im Blockchain-Netzwerk gehört entweder zur Gruppe der Full Nodes oder ist als Light Node beteiligt. Full Nodes zeichnen sich durch das Halten einer vollständigen Kopie der Blockchain und der Prüfung der Integrität der Daten aus. Daher werden üblicherweise leistungsfähige Rechner genutzt, die sowohl über genügend Speicherkapazität verfügen als auch die notwendige Rechenleistung aufbringen können [72].

Rechenleistung wird in erster Linie für die Berechnung des Konsenses benötigt, die für jeden neuen zu schreibenden Block durchgeführt werden muss. Da bei einer Private Blockchain vorher festgelegt wird, wer am Netzwerk teilnehmen darf, kann die Konsensbildung einfacher und somit weniger rechenintensiv sein als bei Public Blockchains. Jeder Full Node verfügt über eine digitale Unterschrift, mit der jeder neue Block signiert wird, um die Nachvollziehbarkeit der Herkunft zu gewährleisten. Die Wahrung der Konsistenz der bereits in die Blockchain eingefügten Blöcke ist ebenfalls eine Aufgabe der Full Nodes. Im Gegensatz zur Prüfung des Konsens ist diese Berechnung durch die Verkettung von Hash-Werten über Merkle-Trees sehr effizient durchzuführen und benötigt nur wenige Ressourcen [206].

Light Nodes halten nur Hash-Werte von Blöcken, die sie selbst betreffen, und interagieren nicht direkt mit der Blockchain. Sie verfügen jedoch über eine digitale Identität, die sie dazu berechtigt, Daten an Full Nodes zu senden, die in die Blockchain eingetragen werden sollen [72].

Neben den bereits erwähnten Konsensmechanismen, welche die Korrektheit der Daten gewährleisten, wird auch die Reihenfolge der neu zu schreibenden Blöcke festgelegt. Dies ist essenziell, da in der Praxis viele Blöcke gleichzeitig verarbeitet werden müssen.

Nachdem ein Konsens gefunden wurde, wird ein Block geschrieben und an alle Full Nodes im Netzwerk verteilt. Dabei können verschiedenste Konsens-Algorithmen zum Einsatz kommen, die nach Sicherheitsrelevanz ausgewählt werden. Mit steigender Komplexität der Algorithmen erhöht sich sowohl die Sicherheit als auch die benötigte Rechenleistung und Zeit, um einen Konsens zu erreichen. Darüber hinaus lassen sich diese vordefinierten Regeln in den meisten Fällen über zusätzliche, individuell erstellte Regeln, auf die Bedürfnisse des Anwendungsfalls abstimmen.

Datenstruktur einer Blockchainn Um die Manipulationssicherheit der Blockchain zu verstehen, ist es notwendig, sich die zugrunde liegende Datenstruktur zu verdeutlichen. Der Kommunikationsaufwand für einen Abgleich aller Datensätze mit allen Servern im Konsortium, um die Integrität der Daten sicherzustellen, wäre viel zu hoch. Außerdem müsste bei einer Abweichung zuerst ein Konsens mit dem ganzen Netzwerk gefunden werden, um zu identifizieren, welche Daten manipuliert wurden und ersetzt werden müssen.

Um den Kommunikationsaufwand zu minimieren, wurde für die Blockchain eine eigene Art der Datenspeicherung entwickelt. Ein Blockchain-Server sammelt für einen vorher definierten Zeitraum oder eine Anzahl an Transaktionen alle ankommenden Nachrichten, um diese anschließend in einen Datenblock umzuwandeln. Neben den eigentlichen Transaktionen enthält dieser noch einen Zeitstempel, die Signatur des Servers und einen Hash-Wert. Ein Block hat ein vordefiniertes Format, das im ersten Block, dem sogenannten Genesis-Block, festgelegt wird. Der berechnete Hash-Wert ist das Herzstück der Datenstruktur und wird über alle Informationen des vorherigen Blocks gebildet. Dadurch erhält jeder Eintrag in einer Blockchain eine Referenz zum vorherigen Block und es entsteht eine Kette an Blöcken, die eindeutig ist [149].

Um eine Manipulation an einer Transaktion unerkannt durchführen zu können, darf sich durch die Änderung der Hash-Wert des Blocks nicht verändern, da durch eine Änderung des Hash-Wertes die aufgebaute Verkettung verloren geht. Eine Inkonsistenz in der Verkettung führt zu einem deutlich erkennbaren Fehler, welcher automatisch vom Blockchain-Netzwerk bereinigt wird. Ein Abgleich aller Blöcke ist daher nicht mehr notwendig. Die Korrektheit aller Einträge zwischen dem Genesis-Block und dem letzten Block kann nun vom Server selbst überprüft werden. Einzig der letzte Eintrag muss noch mit dem Konsortium abgeglichen werden, um die Integrität der gesamten Blockchain zu garantieren [149].

7.2.2 Smart Contracts

In Smart Contracts können sämtliche Rahmenbedingungen für die Beziehungen der über die Blockchain miteinander verbundenen Partner verbindlich geregelt werden. Smart Contracts bestehen aus Regelwerken, die digital hinterlegt sind und automatisch vom System überwacht werden. Blockchains werden dadurch mehr als nur verteilte und manipulationssichere Datenspeicher, sie ermöglichen die Automatisierung von Prozessen, Regularien und Organisationsprinzipien [184].

Bei Smart Contracts handelt es sich nicht um Verträge im herkömmlichen Sinn, es sind vielmehr programmierte Wenn-Dann-Bedingungen. Bei der Durchführung einer Transaktion wird über diese Bedingungen die Konsistenzwahrung sichergestellt und häufig Folgeprozesse angestoßen. Bei der Konsistenzwahrung werden die Transaktionen selbst auf ihre Korrektheit geprüft. Folgeprozesse könnten zum Beispiel die Erstellung einer Rechnung oder die Durchführung einer Zahlung sein. In Verbindung mit der Manipulationssicherheit der Blockchain ist es auch möglich, Verträge im klassischen Sinne, als Vereinbarung zwischen Geschäftspartnern, abzubilden [62].

Alle Entscheidungen, die automatisiert über den Smart Contract getroffen werden, erfolgen im Verbund des gesamten Blockchain-Netzwerks. Durch die Dezentralisierung von Entscheidungen kann auf eine dritte, überwachende Instanz somit verzichtet werden [127]. Durch die in der Blockchain vorhandenen Einträge, die von den Smart Contracts als Grundlage genutzt werden, sind alle Informationen von den beteiligten Parteien jederzeit einsehbar und damit nachvollziehbar. Die dafür notwendigen Regelwerke können entweder on-Chain (siehe Abschnitt 7.2.2.1) oder off-Chain (siehe Abschnitt 7.2.2.2) hinterlegt werden. Beide Varianten haben Vor- und Nachteile und müssen je nach Anwendungsfall genauer betrachtet werden, um zu entscheiden, welche Variante zur Anwendung kommt.

Als Alternative zu on-Chain und off-Chain kann eine Mischung beider Hinterlegungsarten realisiert werden. Dabei werden einfach umzusetzende Entscheidungen direkt auf der Blockchain implementiert und alles, wofür komplexe Verfahren notwendig sind, wird in separate Programme ausgelagert.

7.2.2.1 On-Chain Smart Contracts

Als on-Chain werden hinterlegte Smart Contracts bezeichnet, wenn die Logik direkt auf der Blockchain umgesetzt wurde. Dabei wird auf die von der Blockchain-Lösung angebotene Sprache zurückgegriffen. Eine Entwicklung direkt auf der Blockchain hat den großen Vorteil, dass Änderungen nicht ohne weiteres vorgenommen werden können. Außerdem liegen die in den Smart Contracts verwendeten Regelwerke für alle sichtbar vor. Wenn sichergestellt wurde, dass ein Smart Contract korrekt arbeitet, kann dieser vollkommen autonom ausgeführt werden und benötigt keine weitere Überwachung [127].

Ein Nachteil von on-Chain entwickelten Smart Contracts ist, dass der produzierte Code nicht mit anderen Blockchain-Lösungen kompatibel ist. Viele Blockchain-Anbieter haben ihre eigene Programmiersprache entwickelt, die mitunter nicht sehr intuitiv und effizient zu handhaben ist. So kann bereits die Implementierung von einfachen Smart Contracts mit sehr viel Aufwand verbunden sein.

Einer der aktuell bekanntesten Anbieter für on-Chain entwickelte Smart Contracts ist die Ethereum-Plattform mit ihrer eigens entwickelten Sprache Solidity. Solidity ist eine Turing-vollständige Programmiersprache, die Ähnlichkeiten zu JavaScript aufweist. Als Ausnahme sollte an dieser Stelle noch die Hyperledger Fabric erwähnt werden, die auf die Entwicklung einer eigenen Programmiersprache verzichtet hat und stattdessen eine On-Chain-Programmierung über verschiedene Hochsprachen wie Go und Java unter dem Begriff „Chaincode“ anbietet [184]. Dabei ist zu beachten, dass das Programmierparadigma „Convention over Configuration“ zur Anwendung kommt und bei der Entwicklung die von Hyperledger Fabric vorgegebenen Konventionen eingehalten werden müssen.

Ein Problem stellt allerdings die noch junge Blockchain-Technologie dar; beinahe täglich kommen neue Plattformen und Konzepte auf und Sicherheitslücken von bestehenden Plattformen werden bekannt. Auch die vermeintlich großen Plattformen wie Ethereum sind nicht vor Angriffen gefeit, wie der DAO-Hack eindrucksvoll bewiesen hat [9].

Neben den großen Möglichkeiten, die die On-Chain-Entwicklung bietet, geht die Agilität, um auf unvorhersehbare Ereignisse wie diese reagieren zu können, verloren.

7.2.2.2 Off-Chain Smart Contracts

Bei der Off-Chain-Variante ist die Entscheidungslogik in ein externes Programm ausgelagert, wodurch Smart Contracts unabhängig von der eingesetzten Blockchain-Lösung entwickelt werden können. Wenn das zugrundeliegende Regelwerk von der Blockchain separiert ist, kommt die Blockchain als dezentraler, manipulationssicherer Datenspeicher zum Einsatz [127].

Für die Implementierung der Smart Contracts kann eine beliebige, allgemeine, höhere Programmiersprache gewählt werden, die bereits etabliert ist und über viele Jahre weiterentwickelt wurde. Durch die freie Wahl der Sprache und die ausgereifte Syntax existiert keine Einstiegshürde in die Welt der Smart Contracts. Es sind viel komplexere Regelwerke ohne großen Aufwand umsetzbar.

Auch wenn einige Vorteile aus dem direkten Zusammenspiel von Smart Contracts und Blockchains verloren gehen, bleiben die Integrität und die Dezentralität der Daten erhalten. Durch die Integrität und Dezentralität kann ein hoher Grad an Automatisierung erreicht werden, aber vollkommen autonom laufende Prozesse sind nicht möglich. Es muss weiterhin eine Überwachung erfolgen, da Änderungen am Regelwerk und somit am Smart Contract jederzeit möglich sind.

7.2.3 „Internet of Things“ auf Landmaschinen

Im Kontext des Internet of Things wird eine immer stärkere Digitalisierung von Industrieprozessen angestrebt. Im Zuge dieser Digitalisierung werden immer mehr Maschinen und andere produktionsrelevante Ressourcen als cyber-physische Systeme betrachtet [14].

> Cyber-Physical Systems adressieren die enge Verbindung eingebetteter Systeme zur Überwachung und Steuerung physikalischer Vorgänge mittels Sensoren und Aktuatoren über Kommunikationseinrichtungen mit den globalen, digitalen Netzen. Dieser Typus von Systemen ermöglicht über Wirkketten eine Verbindung zwischen Vorgängen der physischen Realität und den heute verfügbaren, digitalen Netzinfrastrukturen. [52]

Auch moderne Landmaschinen entsprechen dieser Definition. Sie sind bereits hochgradig technologisiert und spielen eine wichtige Rolle in der digitalisierten Landwirtschaft. Insbesondere die Erfassung von Telemetriedaten auf Erntemaschinen und Traktoren sowie deren Anbaugeräten hat sich etabliert [202]. Durch eine Vielzahl von Sensoren werden die internen Zustände der Maschinen erfasst, z. B. Motorlasten und Kraftstoffverbrauch oder die aktuelle Ernteleistung eines Feldhäckslers [154].

Fusioniert mit GPS-Daten entstehen so geo-referenzierte Datenströme, die relevante Kennwerte der landwirtschaftlichen Arbeit enthalten. Übermittelt werden die Daten zumeist an eine zentrale Verarbeitungsstelle, beispielsweise Farm Management Information Systems (FMISs). Diese ERP-Systeme der Landwirtschaft sind oftmals als Cloud-Lösungen implementiert und bieten Werkzeuge, um die betrieblichen Abläufe eines landwirtschaftlichen Betriebs oder eines Lohnunternehmens zu planen [95, 121].

Schon heute unterstützen diese Systeme den Landwirt bei der Abwicklung von Geschäftsprozessen. So wird zum Beispiel durch die automatische Generierung von Arbeitsdokumentationen die manuelle Erstellung von Rechnungen erleichtert. Hierbei handelt es sich aber stets um eine nachgelagerte Datenverarbeitung, die nicht im laufenden Arbeitsprozess genutzt werden kann. Ebenso wenig lassen sich die Beziehungen zwischen verschiedenen Geschäftspartnern digital abbilden. Es werden lediglich digitalisierte Schnittstellen bereitgestellt, die es erlauben, potenzielle Geschäftspartner zu verknüpfen, z. B., um direkt über eine FMIS-Plattform neues Saatgut für die nächste Ernte zu bestellen.

7.2.4 Zahlungsverfahren

Für einen vereinfachten Finanzfluss entlang der Supply-Chain werden bargeldlose Zahlungsverfahren benötigt, die einen hohen Automatisierungsgrad unterstützen und von möglichst vielen Finanzinstituten akzeptiert werden. Ein weiterer wichtiger Faktor ist eine hohe Akzeptanz der Zahlungsverfahren im B2C- und B2B-Bereich.

Laut dem Bundesverband deutscher Banken teilen sich die bargeldlosen Zahlungsverfahren in Lastschrift, Überweisung, Kreditkarten, Debitkarten, E-Geld, Schecks und sonstige auf. Trotz der in den letzten Jahren steigenden Beliebtheit von E-Geld (z. B. PayPal, paydirekt, Bitcoin etc.) hat dieses nur einen Anteil von 0.2 % an den bargeldlosen Transaktionen. Vorherrschend in Deutschland ist die Lastschrift mit 50,6 %, gefolgt von der Überweisung mit 29,6 % [53]. Lastschriften und Überweisungen müssen bei einer Bank eingereicht werden. Auf dem digitalen Weg kann dies mit den Standards FinTS und EBICS erreicht werden. FinTS findet im Privatkundenbereich Anwendung, wohingegen EBICS eher für das geschäftliche Umfeld konzipiert wurde. Beide Standards bedienen sich asymmetrischer Verschlüsselungstechnologien und kommen nach dem initialen Schlüsselaustausch mit dem Kreditinstitut ohne die manuelle Eingabe von zusätzlichen Sicherheitsmerkmalen, beispielsweise TAN oder PIN, aus und eignen sich deshalb besonders für eine automatisierte Verarbeitung von Finanztransaktionen. FinTS wurde im Jahr 2004 von mehr als 2000 von 2401 Kreditinstituten in Deutschland unterstützt. EBICS ist verpflichtend, d. h., alle deutschen Kreditinstitute müssen das EBICS-Protokoll für die Kontoführung anbieten. Im Jahr 2016 waren das insgesamt 1888 Institute [54]. Des Weiteren bietet die Mehrheit der französischen und Schweizer Banken EBICS an [90].

Mit dem europaweiten Inkrafttreten der Zahlungsdiensterichtlinie 2 (Payment Service Directive 2 (PSD2)) im Januar 2018 sind Banken verpflichtet, ihre Schnittstellen für Drittanbieter zu öffnen. Die Richtlinie schafft dafür die nötigen rechtlichen Grundlagen und beschert dem Kunden eine größere Freiheit bei der Auswahl von Finanzdienstleistungen, da nun Drittanbieter durch Autorisation des Kunden auf dessen Daten zugreifen dürfen. Eine Herausforderung für die Finanzdienstleister sind die teils proprietären Schnittstellen der Banken. Hier könnte das NextGenPSD2-Framework der Berlin Group als eine allgemeine und europaweite PSD2-Schnittstellenspezifikation Abhilfe schaffen [211]. Die Berlin Group ist eine europaweite Initiative für Interoperabilitätsstandards und Harmonisierung im Zahlungsverkehr. Ihr gehören unter anderem Mastercard, VISA, Die Deutsche Kreditwirtschaft und EURO-Kartensysteme an.

Zunehmend an Bedeutung gewonnen haben in den letzten Jahren digitale Geldbörsen. Ein Vertreter dieser Gruppe ist PayPal und bietet neben einem Online-Portal und diversen Applikationen für den Desktop- und Mobile-Bereich auch Online-Schnittstellen, über die die angebotenen Dienste in die eigene Umgebung integriert werden können. Es ist möglich, sowohl B2C- als auch B2B-Funktionalitäten zu verwenden. PayPal fungiert als Zwischenhändler. Das bedeutet, Geldbeträge, die mittels PayPal transferiert werden, belasten ein vorher hinterlegtes Girokonto per Lastschrift oder eine Kreditkarte des Senders. Paydirekt ist die deutsche Konkurrenz zu PayPal, bietet allerdings nur eine programmatische Schnittstelle für Händler, die auf den Bestellprozess in einem Online-Shop zugeschnitten ist (B2C). Das Transferieren von Geldbeträgen ist nur über die offizielle paydirekt-Mobile-Applikation möglich.

Die Kryptowährung Bitcoin verwendet, ebenso wie FinTS und EBICS, Schlüssel für die Signierung von Transaktionen. Damit Transaktionen in die Blockchain -– eine verteilte Datenstruktur und sozusagen das digitale Kontobuch der Kryptowährung -– übernommen werden können, müssen diese von sogenannten Minern durch Lösung eines kryptografischen Rätsels berechnet werden. Die Miner werden für ihren Aufwand durch die Blockchain selber in der Kryptowährung belohnt, allerdings reicht diese Belohnung heutzutage nicht mehr aus, wenn man seine Transaktionen zeitnah gespeichert haben will. Man muss eine zusätzliche Vergütung ausloben, was zu insgesamt sehr hohen Transaktionsgebühren führen kann [150]. Zeitnahe Transaktionsverarbeitung bei Bitcoin bedeutet beispielsweise, dass eine Transaktion im Durchschnitt nach 10 min gespeichert werden kann [150]. Aufgrund des Konsensmechanismus der Blockchain kann es jedoch passieren, dass die zuvor gespeicherte Transaktion als ungültig markiert bzw. verworfen wird. Je länger eine Transaktion in der Blockchain verweilt, desto höher ist die Wahrscheinlichkeit, dass sie von allen Beteiligten akzeptiert wird (Konsensmechanismus). Bei Bitcoin kann man nach zirka 60 min davon ausgehen, dass eine Transaktion von der Mehrheit der Beteiligten akzeptiert wurde [150].

7.2.5 Elektronische Rechnungen

Das Forum elektronische Rechnung Deutschland (FeRD) definiert eine elektronische Rechnung als „[...] eine Rechnung, die in einem elektronischen Format ausgestellt, übertragen und empfangen wird“ [31]. Es wird zwischen strukturierten und unstrukturierten E-Rechnungen unterschieden. Zu den strukturierten Rechnungsformaten gehören z. B. EDI, XML, XRechnung und ZUGFeRD 2.0 (Profile EN16931). Unstrukturierte Formate sind unter anderm PDF-, E-Mail-Text-, Bilddateien und Word-Dokumente. Die alte ZUGFeRD-Version 1.0 nimmt eine Sonderstellung ein, da sie strukturierte und unstrukturierte Rechnungsdaten in einem hybriden Format kombiniert [31].

Mit dem deutschen E-Rechnungsgesetz werden die europäischen Vorgaben – definiert in der Richtlinie 2014/55/EU – in nationales Recht umgesetzt. Das Gesetz tritt am 27. November 2018 verbindlich für den Großteil der öffentlichen Auftraggeber in Deutschland in Kraft. Für Unternehmen, die Dienstleistungen oder Güter gegenüber öffentlichen Vergabestellen in der Europäischen Union abrechnen, hat dies direkte Auswirkungen, denn sie müssen Rechnungen in naher Zukunft – spätestens bis November 2019 – in einem strukturierten elektronischen Format einreichen.

Aber auch für alle anderen Unternehmen kann es lohnenswert sein, auf eine elektronische Rechnungsstellung (E-Invoicing) umzustellen, um von den erheblichen Einsparpotenzialen durch die Vereinfachung unternehmensinterner Prozesse und durch die Reduzierung von Druck-, Papier- und Portokosten zu profitieren [126]. Die strukturierten Rechnungsformate, die der EU-Richtlinie genügen, können sich hierbei als adäquat erweisen. Dies sind beispielsweise XRechnung und ZUGFeRD in der Version 2.0.

7.3 Smart Contracting in der Landwirtschaft

Viele Arbeiten in der Landwirtschaft erfordern die Kooperation mehrerer Betriebe, da ein einzelner landwirtschaftlicher Betrieb meist nicht alle nötigen Ressourcen (z. B. Maschinen, Anbauflächen, Zeit, Arbeitskraft, Fachkenntnisse etc.) hält, die für eine Vielzahl spezialisierter Arbeitseinsätze nötig sind. Der Einsatz von Lohnarbeit und Nachbarschaftshilfe prägen die landwirtschaftliche Arbeitswelt, um den wechselnden Anforderungen gerecht zu werden.

Oftmals werden Kooperationen spontan abgesprochen und nach geleisteter Arbeit in Rechnung gestellt. Eine vorab getroffene vertragliche Vereinbarung besteht somit selten. Im Hinblick auf die starke Abhängigkeit von Witterungsverhältnissen hat sich dieses Vorgehen über Jahrhunderte bewährt, steht aber heutzutage im Kontrast zu den immer stärker werdenden rechtlichen Auflagen und Anforderungen moderner Landwirtschaft.

So muss beispielsweise im Zuge der neuen Düngemittelverordnung die Ausbringung von Gülle und anderen Düngern sehr genau geplant und dokumentiert werden. Bei Nichtbeachtung der festgeschriebenen Grenzwerte drohen strenge Sanktionen für den Landwirt (LW). Änderungen wie diese verschärfen den Bedarf an vertraglich geregelten Kooperationen, um die Zuständigkeiten und Leistungsumfänge von landwirtschaftlicher Lohnarbeit im Vorfeld zu fixieren und eine entsprechende Rechtssicherheit zu gewährleisten. Die Definition von Service-Level-Agreements, wie es in anderen Industrien (z. B. der Logistik) schon gängige Praxis ist, gewinnt so auch in der Landwirtschaft immer mehr an Bedeutung. Der Aufwand, der mit dem Nachverfolgen ebendieser Agreements verbunden ist, übersteigt jedoch zumeist die Möglichkeiten der Kooperationspartner, sodass auf allgemeine AGB zurückgegriffen wird und die speziellen Anforderungen des aktuellen Auftrags lediglich mündlich abgesprochen werden.

Digitale Verträge und deren automatische Nachverfolgung bieten somit viele Chancen in der Landwirtschaft. Deshalb wurde die Produktion von Silomais als Gärsubstrat für Biogasanlagen als Praxisbeispiel ausgewählt, um den in diesem Artikel beschriebenen Smart-Contracting-Ansatz zu testen.

7.3.1 Landwirtschaftlicher Anwendungsfall

Als landwirtschaftlichen Prozess betrachten wir die Ernte von ganzen Maispflanzen mit einem selbstfahrenden Feldhäcksler und den anschließenden Transport des Ernteguts zur Lagerstätte mit einer Flotte von Transportfahrzeugen sowie die anschließende Produktion von Maissilage an der Lagerstätte durch Aufschieben und Verdichten des gehäckselten Maises in einem Fahrsilo.

Als betriebswirtschaftlichen Prozess legen wir den Kontext einer gewerblichen Biogasanlage (BGA) zugrunde, weil hier die betriebswirtschaftlichen Beziehungen teilweise stark von der eigentlichen Arbeitsorganisation abweichen, wie in Abbildung 7.1 dargestellt ist.

Der Betreiber einer BGA benötigt über das Jahr hinweg eine gewisse Menge an Gärsubstrat für seine Anlage. Dafür wird einmal im Jahr eine Maissilage ausreichender Größe angelegt, wodurch ein signifikanter landwirtschaftlicher Aufwand entsteht, den der Betreiber der BGA im Regelfall nicht selbst leistet, sondern an Subunternehmen auslagert. Dies gilt insbesondere dann, wenn die BGA gewerblich und nicht durch einen landwirtschaftlichen Betrieb betrieben wird.

Der Anbau der Maispflanzen wird auf eine Vielzahl von Landwirten ausgelagert. Diese bestellen ihre eigenen Felder, pflegen die Pflanzen bis ins geeignete Reifestadium und verkaufen anschließend die fertigen Maispflanzen an den Betreiber der BGA, welcher sie als Grundlage für das Gärsubstrat nutzt. Üblicherweise wird der geerntete Mais pro Tonne Trockenmasse abgerechnet. Um eine korrekte Abrechnung zu ermöglichen, muss jede Lieferung, die am Silo ankommt, dem richtigen Feld und damit dem LW zugeordnet werden. Ebenso wird durch eine Fahrzeugwaage bestimmt, wie viel Nettogewicht Maishäcksel ein Transportfahrzeug am Silo abgeladen hat. Darüber hinaus werden Proben des Ernteguts entnommen, um in nachgelagerten Labortests den Trockenmassegehalt des Materials zu bestimmen. Die erhobenen Daten geben Auskunft über die Menge und Qualität des Ernteguts und sind ausschlaggebend für die Höhe der Bezahlung.

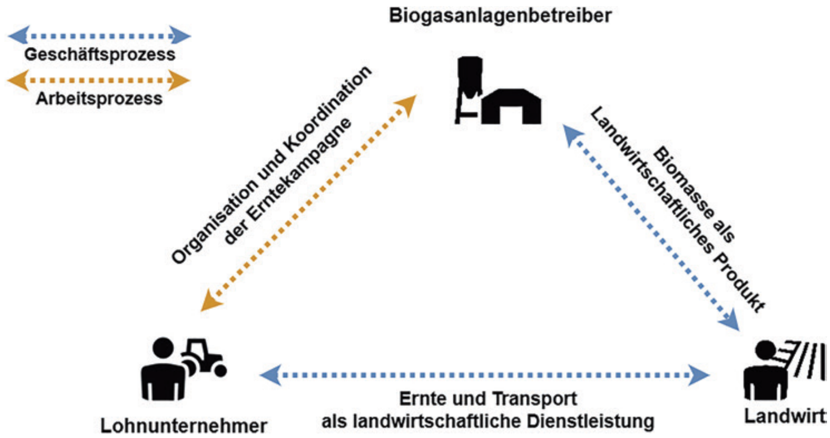


Figure 7.1: Betriebswirtschaftliche Beziehungen im Anwendungsfall

Die Leistungsabnahme geschieht manuell und es müssen aufwendige Kontrollmechanismen (z. B. ein Vieraugenprinzip) integriert werden, um eine fehlerfreie Dokumentation zu gewährleisten. Das Führen eines „Wiegeprotokolls" generiert somit erheblichen Aufwand für den Betreiber der BGA und steht im Konflikt mit den anderen Aufgaben, die zur Prozesslaufzeit anfallen, wie die Disposition der Abfuhrlogistik. Ebenfalls erfordert dieses Vorgehen ein hohes Maß an Vertrauen des LW gegenüber dem Betreiber der BGA, da die Dokumentation von Mitarbeitern der BGA durchgeführt wird. Die Leistungsdokumentation liegt also beim Leistungsnehmer und ist für den Leistungserbringer, den LW, nicht vollständig transparent.

Die Durchführung der Ernte sowie die Produktion der Silage werden ebenfalls vergeben. Ein landwirtschaftlicher Lohnunternehmer (LU) stellt die nötigen Maschinen sowie die Arbeitskräfte. Oftmals wird auch die Organisation der Erntelogistik über alle Felder der zubringenden LW an den Disponenten des LU ausgelagert. Der LU steht also in engem Kontakt mit dem BGA-Betreiber und den individuellen LW. Um die Leistungen, die das LU erbringt, mit der BGA abzurechnen, werden die geleisteten Arbeitsstunden der Maschinen und Fahrer erfasst und je nach Abrechnungsart werden auch die bearbeiteten Flächen, die gefahrenen Kilometer sowie der verbrauchte Kraftstoff protokolliert und abgerechnet. Diese Dokumentation übernehmen die Mitarbeiter des LU, welche die Landmaschinen fahren und die Ernte durchführen. Es ergeben sich die gleichen Probleme mit der Leistungsabnahme wie zuvor. Sie ist intransparent, weil der Leistungserbringer seine eigene Leistung dokumentiert und fehleranfällig, weil die Dokumentation neben der eigentlichen landwirtschaftlichen Arbeit oft vernachlässigt wird.

Der landwirtschaftliche Arbeitsprozess legt es nahe, dass der LU dem Betreiber der BGA den Gesamtaufwand für die Silomaisernte und -produktion in Rechnung stellt, so wie auch die LW für die Produktion der Biomasse eine Rechnung an den BGA-Betreiber stellt. Auf betriebswirtschaftlicher Ebene wird dies jedoch oftmals aktiv vermieden, da eine direkte Beauftragung des LU durch die gewerbliche BGA zur Folge hat, dass die Arbeiten des LU nicht als landwirtschaftliche Arbeit, sondern als gewerbliche Arbeit klassifiziert werden. Dies verhindert die Inanspruchnahme von Agrarsubventionen (bspw. auf Diesel) und verschärft die Auflagen bei der Durchführung der Ernte, z. B. im Hinblick auf zulässige Arbeitszeiten und die Verwendung von Fahrtenschreibern.

Um die wirtschaftlichen Vorteile der landwirtschaftlichen Arbeit adäquat auszunutzen, rechnet der LU deshalb mit den individuellen LW direkt ab, da es sich so um eine landwirtschaftliche Dienstleistung handelt. Die LW wiederum legen die so entstehenden Kosten für Ernte und Transport auf den Betreiber der BGA um, indem sie in die Produktionskosten der Biomasse einbezogen werden. Der LW verkauft also keine ganzen Maispflanzen auf dem Feld, sondern vorproduzierten Maishäcksel, inklusive Anlieferung zum Silo. Dies gilt genauso als landwirtschaftliches Produkt, als würde er den Mais „ab Feld“ verkaufen.

Diese beschriebene Art der Abrechnung ist zwar wirtschaftlicher für alle beteiligten Parteien, entkoppelt aber auf betriebswirtschaftlicher Ebene die Geschäftspartner, die im landwirtschaftlichen Arbeitsprozess sehr eng miteinander arbeiten. Hierdurch entsteht, wie beschrieben, ein erheblicher Mehraufwand bei der Dokumentation und der Abrechnung. Ebenso entstehen durch die manuelle Abrechnung der Ernte erhebliche Verzögerungen im Geldfluss. Die Rechnungstellung erfolgt heute noch auf Papier und zur Zahlung werden SEPA-Überweisungen genutzt. Zwischen Leistungserbringung und Abschluss aller damit verbundenen Zahlungen vergehen üblicherweise mehrere Wochen.

Hier können Ansätze des Smart Contracting helfen. Zunächst müssen die betriebswirtschaftlichen Abhängigkeiten zwischen den Geschäftspartnern im Vorfeld über digitale Verträge abgebildet werden. Anschließend muss die manuelle Leistungsabnahme abgeschafft und auf eine digitale Übermittlung direkt von den Landmaschinen, in einer für beide Partner transparenten Dokumentation implementiert werden.

Werden die Geschäftsprozesse digital beschrieben und Leistungsabnahme automatisiert durch ein technisches System bereitgestellt, können sich die Akteure der Silomaisernte wieder voll auf die Durchführung der Erntekampagne konzentrieren. Es ergeben sich jedoch noch weitere Vorteile. Aufbauend auf der automatischen Leistungsabnahme lässt sich eine Business-Logik implementieren, die es erlaubt, die festgelegten Leistungen zu prüfen. So können Abweichungen von den vereinbarten Verträgen schon zur Laufzeit der Ernte aufgedeckt und behandelt werden. Ebenso lassen sich nachgelagerte Arbeitsschritte wie die Abrechnung zwischen den Geschäftspartnern automatisieren.

7.3.2 Referenzszenario

Im Rahmen des Projekts SOFiA wurde 2017 eine Erntekampagne für eine BGA begleitet. Über 5 Erntetage wurden rund 12.000 Tonnen Silomais eingefahren, die die BGA in 2.2 Mio. Kubikmeter Biogas umsetzt. Dieses wird anschließend in einem Blockheizkraftwerk umgesetzt, wodurch sich jährlich je 4.8 Mio. kWh an Strom und Wärme ergeben.

Die Ernte umfasste 60 Felder mit einer Gesamtfläche von 250 Hektar, bereitgestellt von 15 LW. Es kamen zwei Feldhäcksler mit unabhängigen Transportketten zu je fünf Transportfahrzeugen zum Einsatz. Zusätzlich wurden zwei Traktoren für die Silageproduktion eingesetzt.

Die Erntekampagne wurde, wie eingangs beschrieben, dokumentiert und abgerechnet. Das Wiegeprotokoll mit etwa 900 individuellen Einträgen wurde in MS-Excel erfasst. Zusätzlich wurde eine handschriftliche Kontrollliste geführt. Diese Arbeiten wurden durch den Disponenten der Biogasanlage durchgeführt.

Nach der Ernte wurden die individuellen Daten der LW separiert und an die 15 LW übersandt, welche wiederum auf Basis dieser Daten individuelle Rechnungen erstellten. Nachdem diese an den Betreiber der BGA zurückgesandt wurden, wurden die Rechnungen per Überweisung beglichen. Analog dazu stellte der LU manuell 15 Rechnungen an die LW. Es wurden also insgesamt 30 Rechnungen durch 17 verschiedene Betriebsleiter erstellt, obwohl die relevanten Daten ursprünglich schon gebündelt vorlagen. Der so entstehende bürokratische Mehraufwand ist durch die höhere Wirtschaftlichkeit des Vorgehens zwar gerechtfertigt, aber dennoch unnötig kompliziert. Dies gilt insbesondere, wenn man bedenkt, dass alle relevanten Informationen aus der landwirtschaftlichen Arbeit entnommen werden und durch eine technische Prozessbetrachtung automatisiert erfasst und weiterverarbeitet werden können.

7.4 Smart-Contract-Plattform

Aufgrund der Vielzahl an unterschiedlichen Lösungen wurde die komplette Business-Logik in eine Smart-Contract-Plattform ausgelagert. Es ist aktuell noch nicht ersichtlich, welche Lösung sich am Markt durchsetzen wird. In einem späteren Roll-out des Pilotprojekts ist geplant, einfache Smart Contracts auch on-Chain zu halten. Ein Großteil der Business-Logik wird allerdings weiterhin unabhängig bleiben, um die Portabilität der Software zu gewährleisten.

Die Smart-Contract-Plattform ist eine webbasierte Anwendung, die von allen Parteien des Blockchain-Konsortiums genutzt werden kann. Neben dem Service, der das Regelwerk für den Prozessablauf auswertet, besteht eine zentrale Aufgabe darin, die gesammelten Daten aufzuarbeiten. Alle aktuell laufenden Verträge mit den dazugehörigen Regelwerken und automatisch getätigten Entscheidungen werden in Echtzeit allen Vertragspartnern über eine Graphical User Interface (GUI) zur Verfügung gestellt. Die Smart-Contract-Plattform besteht aus zwei interagierenden Layern, welche die Ebene der Geschäftsprozesse und die der Arbeitsprozesse repräsentieren. Zudem bietet sie die Möglichkeit, weitere Dienstleistungssysteme anzubinden und ist somit der zentrale Knotenpunkt des Systems. So ist beispielsweise eine digitale Plattform zur automatischen Rechnungserstellung und Abwicklung von Finanztransaktionen Teil der SOFiA-Architektur.

Die Ebene der Geschäftsprozesse dient der Repräsentation und Abwicklung von Geschäftsprozessen auf Basis digitaler Verträge und deren Business-Logik. Hier verknüpft die SOFiA-Architektur eine Vielzahl zentraler Plattformen zu einem Netzwerk von Services rund um die logistischen Dienstleistungen. Die Smart-Contract-Plattform stellt eine Schnittstelle bereit, über die Geschäftspartner die Leistungsumfänge einer logistischen Dienstleistung verhandeln und festlegen können. Aus den ausgehandelten digitalen Verträgen werden die notwendigen

Regelwerke extrahiert und Smart Contracts erzeugt. Mithilfe der Smart-Contract-Plattform werden die Smart Contracts verwaltet und auf Basis einer anwendungsspezifischen Business-Logik automatisiert geprüft und bearbeitet. Dazu sind Informationen aus dem Arbeitsprozess notwendig, die im laufenden Betrieb erhoben und bereitgestellt werden.

Die Ebene der Arbeitsprozesse dient der sensorischen Erfassung von Events in einer Logistikkette. Sie besteht aus einem dezentralen Netzwerk aus CPS, auch Smart Objects genannt. Diese erlauben es, die Ressourcen einer logistischen Kette so zu digitalisieren, dass diese ihre Rolle im Prozess verstehen und ihren jeweils eigenen Arbeitszustand zur Prozesslaufzeit ermitteln können. Sie bieten somit die Datengrundlage, um die in einem Smart Contract festgeschriebenen Leistungen zu prüfen.

Im Rahmen des Projekts SOFiA werden die Smart Objects ebenfalls genutzt, um logistische Prozesse auf quantitativer Basis unmittelbar auf der Prozessebene zu steuern. Hierdurch sollen schnellere Reaktionen auf Problemfälle ermöglicht und die Effizienz der logistischen Kette gesteigert werden. Für Details zum dezentralen Planungs- und Steuerungsansatz und die Prozessbetrachtung zur Laufzeit siehe [1], [79] und [225, 226].

7.4.1 Technische Umsetzung

Abgebildet wurde das Architekturkonzept auf ein Netzwerk aus Blockchain-Servern. Der Einsatz von Blockchain-Technologie wurde gewählt, da bei einer Umsetzung von Smart Contracts mit anderen Technologien wie Datenbanken immer ein Vertrauensproblem vorherrscht. Die Hoheit über die Daten liegt üblicherweise bei einem Partner, der administrativen Zugriff hat. Dieses Vertrauensproblem kann ohne die Blockchain- Technologie nur über die Beauftragung einer dritten überwachenden Partei gelöst werden.

Insbesondere Private Blockchains eignen sich durch die Speicherung der Daten auf den lokalen Unternehmensservern besonders gut für den Einsatz im industriellen Umfeld. Für die Smart Contracts werden neben den Prozessinformationen auch hochsensible Daten wie Vertragsdetails und Kontoinformationen transparent in der Blockchain gehalten. Kein Unternehmen möchte die Hoheit über die eigenen Daten aufgeben. Eine zentrale Anforderung an die Technologie ist daher, dass jederzeit nachvollziehbar sein muss, wo die erhobenen Daten liegen und wer Zugriff darauf hat.

Eine weitere Möglichkeit, um Transparenz zu erreichen, liegt in der Verteilung der Daten über Cloud-Systeme. Dies geschieht jedoch zulasten der Manipulationssicherheit. Ein weiterer größer Nachteil beim Einsatz von Cloud-Systemen ist, dass die Daten auf den unternehmensfremden Servern der Cloud-Anbieter gespeichert werden [63].

Implementierung der Blockchain Für die Pilotierung der Smart-Contract-Plattform kommt eine Private-Blockchain-Lösung namens MultiChain zum Einsatz. Durch die schnelle Integration von Partnern in das System und die Fähigkeit, einen Datenaustausch über das Blockchain-Netzwerk zu ermöglichen, eignet sie sich besonders gut für diesen Anwendungsfall. Eine zusätzliche Integration von beispielsweise Middleware-Systemen zur Kommunikation ist bei dieser Softwarelösung nicht notwendig.

Dabei nutzt die MultiChain die Bitcoin-Blockchain als grundlegende Technologie und erweitert die vorhandene Funktionalität um das Verarbeiten von Daten. Der Rechenaufwand für den

Konsensmechanismus wird durch die vorher definierte Anzahl an Teilnehmern begrenzt. Somit sind häufige Kritikpunkte, wie der Stromverbrauch oder auch Latenzzeiten der Server, keine Hindernisse für den Einsatz dieser Blockchain-Lösung im vorliegenden Anwendungsfall (siehe Abschnitt 7.2.1).

Die MultiChain wurde speziell für den Austausch von Daten entwickelt und setzt auf eine Strukturierung der enthaltenen Daten. Die abgelegten Blöcke sind nicht mehr nur in einer nach Transaktionseintritt aufgebauten verketteten Liste gespeichert, sondern werden in eine Datenstruktur abgelegt. Bei der Übermittlung einer Transaktion an die Blockchain wird der Ablageort über sogenannte Streams und Keys angegeben. Die Datenblöcke, die innerhalb der Keys abgelegt sind, werden als „data items" bezeichnet, siehe Abbildung 7.2. Durch diese Adressierung ist ein einfacher und schneller Zugriff auf einzelne Bereiche möglich.

Darüber hinaus ist auch eine Einschränkung des Zugriffs über diese Streams möglich; so können Lese- und Schreibrechte für jeden Stream einzeln vergeben werden. Insbesondere kann auf diese Weise verhindert werden, dass Vertragsdaten von Konkurrenzunternehmen eingesehen werden. Zum Beispiel die Payment-Plattform ist zwar als zentraler Dienst im Blockchain-Netzwerk integriert, kann aber nur auf Zahlungsinformationen und Vertragsdaten, die zur Abrechnung notwendig sind, zugreifen.

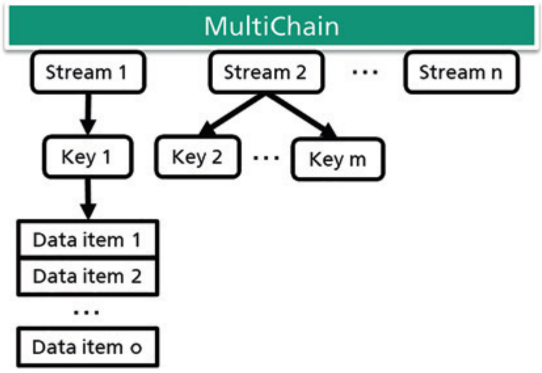


Figure 7.2: Datenstruktur MultiChain

7.4.2 Anwendung auf die Silomaisernte

Für die Silomaisernte besteht ein Vertrag zwischen Landwirt und BGA-Betreiber. Gegenstand dieses Vertrags ist, welche Menge an vorproduzierten Maishäcksel zum Silo geliefert werden muss. Für die Abwicklung des Ernteprozesses besteht ein weiterer Vertrag zwischen Landwirt und Lohnunternehmer. In diesem Vertrag wird geregelt, welches Feld mit welcher Maschine abgearbeitet wird. Zusätzlich müssen Maschinen-, Betriebskosten und benötigte Arbeitszeit für eine spätere Abrechnung aufgenommen werden.

Die einzusetzenden Landmaschinen und das Feld werden bei Vertragsabschluss festgelegt und als Auftrag an die Smart-Contract-Plattform übermittelt. Diese Aufträge können vom Lohnunternehmen abgerufen und über ihre persönliche digitale Identität angenommen werden. Beide Transaktionen gemeinsam ergeben einen digitalen Vertrag. Aus diesem Vertrag wird das

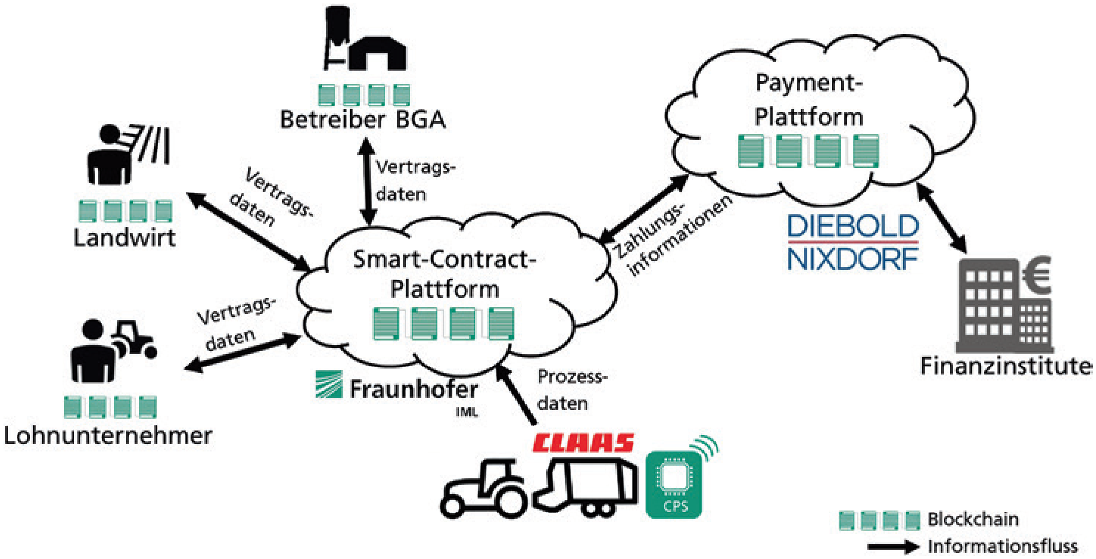


Figure 7.3: Blockchain-Architektur

Regelwerk generiert, das für die Nachverfolgung notwendig ist. Wenn sich nun die Landmaschinen auf dem richtigen Feld befinden, was durch GPS-Daten ermittelt wird, werden alle eingehenden Transaktionen dem entsprechenden Smart Contract zugeordnet.

Vertragsänderungen sind bis zum Abschluss des Vertrags möglich, wobei die Informationen des Originals nicht verändert werden können. Stattdessen muss eine Transaktion über die Aktualisierung, beispielsweise Änderungen des Feldes aufgrund von Witterungsbedingungen, getätigt und von den anderen Vertragsparteien erneut akzeptiert werden.

Sowohl der Biogasanlagenbetreiber und der Lohnunternehmer als auch der Landwirt sind als Full Nodes am Netzwerk beteiligt. Dabei müssen keine Investitionen getätigt werden. Für die notwendige Rechenleistung und Speicherkapazitäten können die bereits vorhandenen Unternehmensserver oder externe Serverressourcen genutzt werden. Um einen automatischen Ablauf der Abrechnung zu realisieren, wird auch der Payment-Plattform Zugriff auf die Vertragsdaten gewährt und diese als Full Node in das Blockchain-Netzwerk integriert. Zur Abrechnung gehören insbesondere die Erstellung der Rechnung über die erbrachte Leistung und die automatische Zahlung bei Einhaltung der vorher definierten Regeln (siehe Abschnitt 7.4.2). Für das Erheben von Telemetriedaten wurden die Erntefahrzeuge mit dezentralen Steuereinheiten ausgestattet, siehe Abbildung 7.3.

Obwohl jeder Partei zu jeder Zeit die gleichen Daten vorliegen, sind die Verantwortlichkeiten für die Aufnahme der Daten zu verteilen. Entlang von Prozessen, die über einen Smart Contract nachverfolgt werden, müssen an mehreren Stellen relevante Daten erhoben werden, um eine automatische Auswertung der Verträge leisten zu können. Diese Daten müssen direkt an die Blockchain-Server gesendet werden, um eine Manipulation der Daten vor der Sicherung durch die Blockchain zu verhindern.

Die Datenaufnahme kann dabei durch manuelle bzw. teilautomatische Eingabe über z.B. Mobile Devices oder automatische Datenaufnahme erfolgen. Um die Datenintegrität zu gewährleisten, muss jedes dieser Geräte über eine digitale Identität verfügen. Eine digitale Identität kann über das Einbinden in das Blockchain-Netzwerk erlangt werden. Dabei wird die Autorisierung initial von den Vertragspartnern vergeben und das Gerät als Light Node in das Blockchain-Netzwerk eingebunden.

Neben den Applikationen für Mobile Devices zur manuellen Eingabe von Daten liegt der Fokus auf dem automatisierten Sammeln von Informationen über Cyber-Physical System (CPS) (siehe Abschnitt 7.2.3). Durch die umfassende Ausstattung der Landmaschinen mit CPS ist eine lückenlose Aufzeichnung des gesamten Ernteprozesses möglich. Zusätzlich zu den Landmaschinen, die dazu genutzt werden, das aktuell abgearbeitete Feld, Arbeitszeiten und genutzten Kraftstoff nachzuhalten, liefert die Waage am Silo die Erntemengen direkt an die Blockchain. Die Waage am Silo liegt im Verantwortungsbereich des Biogasanlagenbetreibers, der für die automatische Aufnahme der Wiegeergebnisse zuständig ist. Somit können alle Aspekte der zwischen den Parteien bestehenden Verträge überwacht und ausgewertet werden. Durch die transparente und manipulationssichere Speicherung der Daten in der Blockchain, kann der gesamte Vorgang automatisch bearbeitet werden, siehe Abbildung 7.4.

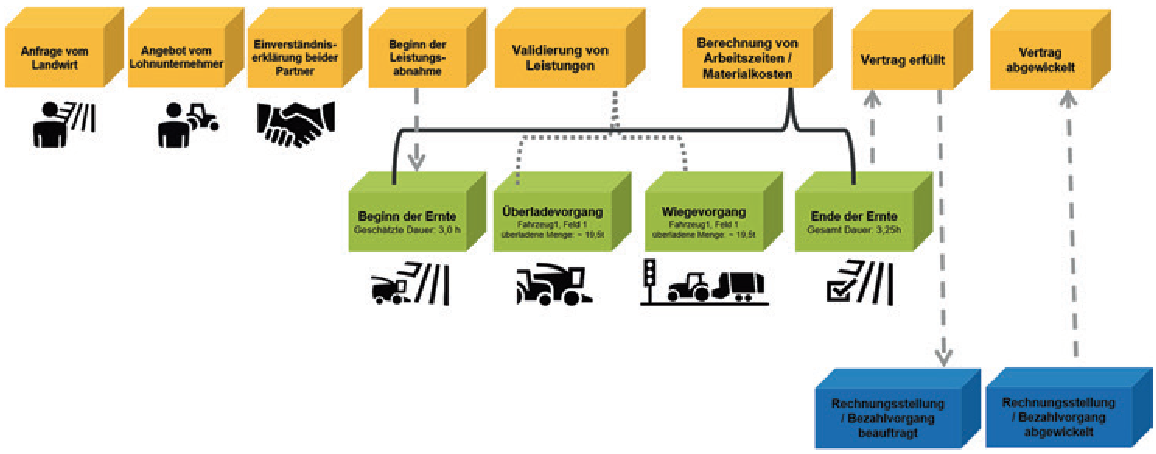


Figure 7.4: Automatisierung Prozess Silomaisernte

Wie bereits im Abschnitt 7.2.2 erläutert, bestehen Smart Contracts aus Wenn-Dann-Bedingungen. Im Folgenden ist in einem kurzen Pseudo-Code-Ausschnitt dargelegt, wie die Abrechnung des Häckselguts am Silo über das digitalisierte Wiegeprotokoll innerhalb des Smart Contracts umgesetzt werden kann.

7.4.3 Integration Payment

Um bei den Vertragspartnern Liquiditätsengpässe zu vermeiden und buchhalterische Aufwände zu minimieren, sind die in Rechnung gestellten Leistungen möglichst zeitnah und automatisiert zu begleichen. Smart Contracts und die Blockchain-Technologie bieten hierfür hervorragende Voraussetzungen. Ist ein Smart Contract erfüllt, wird dies umgehend in der Blockchain durch

```
ALGORITHM Smart-Contract-Anlieferung-Silo

  IF
    Waage mit Hash-Adresse AF38E93F... schickt Tonnage
    AND
    Id des Transportfahrzeugs wird bestätigt
    AND
    Digitale Identitäten sind valide
  THEN
    Geerntetes Feld wird über Latitude und Longitude des
      GPS-Systems am Transportfahrzeug identifiziert
    Weise abgeladene Menge dem zugehörigen digitalen Vertrag zu

    IF
      Bisher abgeladene Menge entspricht der im Contract vereinbarten
      AND
      Die Anlieferung erfolgte innerhalb des vereinbarten Zeitrahmens
    THEN
      Gebe Transaktion frei und stoße damit Abrechnungsprozess an

    END_IF
  END_IF

END_ALGORITHM
```

Figure 7.5: Ein Pseudocode Beispiel eines landwirtschaftlichen Smart Contracts.

die Smart-Contract-Plattform vermerkt und der Bezahlprozess samt Rechnungsstellung kann durch die in Abbildung 7.6 dargestellte Payment-Plattform ausgeführt werden.

Die Payment-Plattform besteht aus einzelnen kleinen Applikationen (Microservices), die dedizierte Aufgaben übernehmen und untereinander über definierte Schnittstellen kommunizieren. Dieser verteilte Ansatz ermöglicht eine Skalierbarkeit, d. h., das gesamte System kann sich dynamisch an Lastsituationen anpassen. Müssen beispielsweise viele Banktransaktionen ausgeführt werden und der dafür zuständige Microservice ist überlastet, startet die Payment-Plattform automatisch weitere Microservices. Konnten alle Transaktionen verarbeitet werden, beendet die Payment-Plattform untätige, nicht mehr benötigte Microservices.

Für die prototypische Umsetzung der Payment-Plattform wurde EBICS als Protokoll beispielsweise als Zahlungsverfahren gewählt, da dieses durch die Verwendung von kryptografischen Schlüsseln relativ einfach automatisierbar ist und von allen deutschen Kreditinstituten unterstützt werden muss (siehe Abschnitt 7.2.4). Für den internationalen Finanzmarkt ist die Unterstützung von weiteren Zahlungsverfahren, zum Beispiel Kreditkartenzahlungen, denkbar.

Kryptowährungen wurden aufgrund der mittlerweile recht hohen Transaktionsgebühren, der starken Kursschwankungen und der langen Transaktionszeiten gegenüber traditionellen unbaren Zahlungsmitteln nicht weiter berücksichtigt.

Automatisierte Transaktionen In den meisten Blockchain-Implementierungen und so auch bei MultiChain werden zu speichernde Daten nicht verschlüsselt, sondern lediglich signiert angefügt. Hierfür wird die digitale Unterschrift des Teilnehmers im Blockchain-Netzwerk (vgl. Full/Light Node Abschnitt 7.2.1) verwendet. Die benötigten Daten für eine Finanztransaktion – bei EBICS sind dies unter anderem IBAN und BIC – sind schützenswert und sollten von anderen Vertragspartnern nicht einsehbar sein. Deshalb wurde ein Datenhaltungskonzept mit einer hybriden Verschlüsselung für die Payment- und Smart-Contract-Plattformen entwickelt. Die sensiblen Transaktionsdaten werden durch ein symmetrisches Verfahren verschlüsselt. Der für das symmetrische Verfahren benötigte Schlüssel wird wiederum durch Einsatz eines asymmetrischen Verschlüsselungsverfahrens geschützt.

Der Ablauf mit Bezug auf Abbildung 7.6 stellt sich wie folgt dar: Der TRX-Data-Provider-Service ist die Schnittstelle der Payment-Plattform zur Blockchain. Er besitzt ein Schlüsselpaar, bestehend aus öffentlichem und privatem Schlüssel. Zu Beginn erfolgt die Bekanntgabe seines öffentlichen Schlüssels über die Blockchain. Die Smart-Contract-Plattform stellt eine Konfiguration für den TRX-Data-Provider-Service zur Verfügung und legt diese mit dem bekannt gemachten öffentlichen Schlüssel verschlüsselt in der Blockchain ab. Die Konfiguration enthält unter anderem den für das symmetrische Verschlüsselungsverfahren zu verwendenden Schlüssel und kann mit dem privaten Schlüssel des TRX-Data-Provider-Service entschlüsselt werden. Steht die Ausführung einer Finanztransaktion an, verschlüsselt die Smart-Contract-Plattform die erforderlichen Daten mit dem Konfigurationsschlüssel und legt sie in der Blockchain ab. Der TRX-Data-Provider-Service liest und entschlüsselt die Transaktionsdaten und übergibt sie über mehrere Warteschlangen an den TRX-Service. Der TRX-Service kommuniziert letztendlich mit dem Kreditinstitut und führt die Transaktion aus. Sowohl das Ergebnis der Transaktionsausführung als auch auch Fehler, die im Ablauf auftreten können, werden symmetrisch verschlüsselt durch den TRX-Data-Provider-Service zurück in die Blockchain geschrieben.

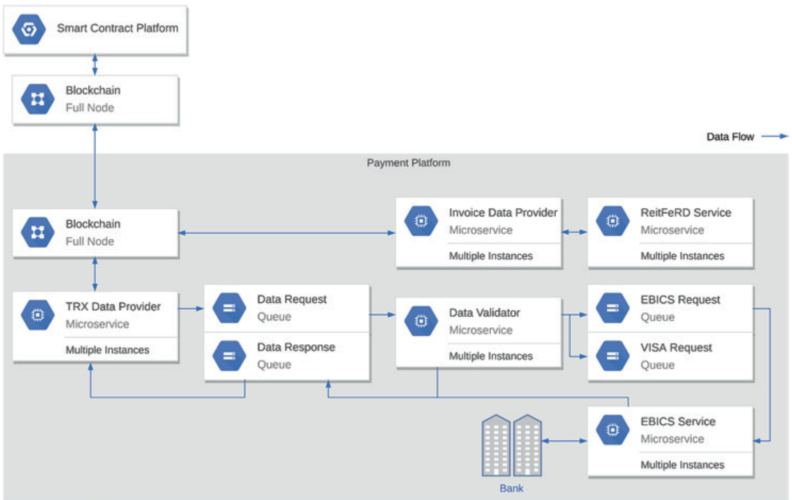


Figure 7.6: Architektur der Payment-Plattform

Automatisierung der Rechnungserstellung Der Invoice-Data-Provider-Service liest für die Rechnungserstellung erforderliche Daten aus der Blockchain und entschlüsselt sie mit dem über die Konfiguration mitgeteilten Schlüssel. Beide Data-Provider-Services – TRX und Invoice – verwenden dasselbe Schlüsselpaar (öffentlicher/privater Schlüssel). Die entschlüsselten Rechnungsdaten werden für die Erstellung einer ZUGFeRD-konformen Rechnung an den ReitFeRD-Service weitergereicht. In einem letzten Schritt wird die erstellte Rechnung durch den Invoice-Data-Provider-Service verschlüsselt zurück in die Blockchain geschrieben.

Abbildung 7.7 zeigt eine mit dem ReitFeRD-Service erstellte Rechnung im PDF-Format.

Kienkie Agrarservice GmbH
Sandkasten Str.80
33804 Hagenstein
Fon 05205 820180
Fax 05205 820181
Mobil 04204 820181
info@agrarservice-kienkie.de

Otmar Dobermann
Dobermann & Sohn GbR
Heringdorfer Straße 108
32138 Penzes
Kundennummer LW0816

Rechnung

Datum: 30.09.2016

Rechnungsnr. 2016004712

Sehr geehrter Herr Otmar Dobermann,

für Erledigung der von Ihnen beauftragten Tätigkeiten berechnen wir Ihnen wie folgt:

Datum	Leistung	Menge	Einheit	Preis / Einh.	Gesamt
25.09.2016	Silomaisernte (inkl. Transport) Schlag: An der Mühle	1,0	ha	220€	220 €
25.09.2016	Diesel	73	l	1,1€	80,30 €
25.09.2016	Silomaisernte (inkl. Transport) Schlag: Vor der Mühle	3	ha	220€	660 €
25.09.2016	Diesel	186	l	1,1€	204,60€

Gesamtsumme, netto	1164,9 €
+ 19% Mehrwertsteuer auf 1164,9 €	221,331
Gesamtsumme, brutto	**1386,231€**

Zahlungsbedingungen:

2,00% Skonto von 1386,231 € = 27,72 €

Zahlbar bis zum 08.10.2016 = 1358,511

Zahlbar ohne Abzug bis zum 30.10.2016

Wir danken für Ihr Vertrauen!

Mit freundlichen Grüßen,

Hendrick Kienkie

Diese Rechnung wurde voll automatisch erstellt und abgerechnet vom SOFiA Smart Accounting System.
Für weitere Informationen besuchen Sie www.sofia-projekt.de/

Kienkie Agrarservice GmbH, Sandkasten Str. 80, 33804 Hagenstein
Bankverbindung: Musterbank Musterstadt, BIC: IBAN:
USt-IdNr. DE 123456789 / Steuernummer 98 765 5432

Figure 7.7: ZUGFeRD-konforme Beispielrechnung

7.5 Diskussion & Ausblick

Der in diesem Beitrag vorgestellte landwirtschaftliche Anwendungsfall zeigt anhand eines praktischen Beispiels, wie die Digitalisierung von Dienstleistungen zwischen Lohnunternehmern, Landwirten und Biogasanlagenbetreibern mithilfe von Blockchains, Smart Contracts und CPS erfolgen kann. Die Maisernte ist durch viele manuelle und wenig formalisierte Prozesse und die Zusammenarbeit verschiedener Akteure (Landwirte, Lohnunternehmer und Biogasanlagenbetreiber) gekennzeichnet. Die Überwachung des Ernteprozesses erfolgt heute noch mit Handzetteln und setzt grundsätzlich großes Vertrauen unter den Akteuren voraus bzw. erfolgt unter dem Vieraugenprinzip.

In diesem Beitrag konnte dargelegt werden, wie mithilfe der Blockchain dieses Vertrauensproblem gelöst werden kann. Durch die Verwendung der Blockchain-Technologie als dezentrales und manipulationssicheres Kommunikations- und Speichermedium erfolgt eine enge Verzahnung der landwirtschaftlichen und betriebswirtschaftlichen Prozesse. Darüber hinaus verdeutlicht der Einsatz von Smart Contracts, wie eine Effizienzsteigerung durch Automatisierung im Ernteprozess erfolgen kann. Die Smart-Contract-Plattform verwaltet und prüft dabei die digitalen Vertragsdaten und überführt sie in Wenn-Dann-Bedingungen, die auch als Smart Contracts bezeichnet werden. Hierfür notwendige Prozessinformationen, beispielsweise Sensordaten oder Arbeitsstunden, werden von CPS geliefert. Die Nachverfolgung, ob und wann ein Feld abgeerntet und ob die richtige Menge beim Silo abgeladen wurde, lässt sich also durch den Einsatz von CPS (Transformation der Landmaschinen, der Gespanne und Waagen zu Smart Objects) ermitteln.

Die Vertrauenswürdigkeit der gelieferten Daten wird über von der Blockchain vergebene, digitale Identitäten gewahrt. Bei Vertragserfüllung werden Rechnungen und Zahlungen durch eine integrierte Payment-Plattform automatisiert erzeugt und durchgeführt.

Wie in diesem Beitrag beschrieben, kann so zum Beispiel das Wiegeprotokoll direkt in der Blockchain gespeichert und dort automatisch mit den Smart Contracts abgeglichen werden, was als Grundlage für eine automatische rechnungslose Transaktion dient.

Die Kombination aus Blockchain, Smart Contract und CPS kann zu einer erheblichen Steigerung der Effizienz im Ernteprozess beitragen. Die gesteigerte Transparenz im Prozess und der Zugewinn an Sicherheit für alle beteiligten Akteure tragen dazu bei. Jedoch setzt dieser technologische Ansatz eine gewisse CPS-Infrastruktur und Konnektivität (Datenaustausch) voraus, die sich auch in der täglichen Praxis bewähren muss. Neben der weiteren Beschreibung von Smart Contracts besteht darüber hinaus das Ziel, die Wirtschaftlichkeit dieses Ansatzes zu bewerten.

References

[6] Alias Abdul-Rahman and Morakot Pilouk. *Spatial data modelling for 3D GIS*. Springer, 2007.

[7] Elcio Abrahão and André Riyuiti Hirakawa. Task ontology modeling for technical knowledge representation in agriculture field operations domain. In *2nd International Conference on Information Systems Engineering (ICISE)*, pages 12–16. IEEE, IEEE, 2017.

[8] Elcio Abrahão and André Riyuiti Hirakawa. Complex task ontology conceptual modelling: towards the development of the agriculture operations task ontology. In *10th International Conference on Knowledge Engineering and Ontology Development (KEOD)*, pages 287–294. INSTICC, SciTePress, 2018.

[9] Dirk Achenbach, Ingmar Baumgart, and Jochen Rill. Die Blockchain im Rampenlicht. *Datenschutz und Datensicherheit*, 41(11):673–677, 2017.

[10] Nikos Alexandratos and Jelle Bruinsma. World agriculture towards 2030/2050: the 2012 revision. Technical report, Food and Agriculture Organization of the United Nations (FAO), 2012.

[11] James F. Allen. An Interval-based Representation of Temporal Knowledge. In *7th International Joint Conference on Artificial Intelligence*, volume 1 of *IJCAI'81*, pages 221–226. Morgan Kaufmann Publishers Inc., 1981.

[12] Carlos Amiama, José M. Pereira, Angel Castro, and Javier Bueno. Modelling corn silage harvest logistics for a cost optimization approach. *Computers and Electronics in Agriculture*, 118(100):56–65, 2015.

[13] Jeff C. Askey. Automated logistic processing and downtime analysis of commercial level multi-pass corn stover harvesting systems. mathesis, Iowa State University, 2014.

[14] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer Networks*, 54(15):2787–2805, 2010.

[15] Hermann Auernhammer. Precision Farming – The Environmental Challenge. *Computers and Electronics in Agriculture*, 30(1-3):31–43, 2001.

[16] Franz Baader. *The description logic handbook: Theory, implementation and applications*. Cambridge University Press, 2003.

[17] Athanasios T. Balafoutis, Bert Beck, Spyros Fountas, Zisis Tsiropoulos, Jürgen Vangeyte, Tamme van der Wal, I. Soto-Embodas, Manuel Gómez-Barbero, and Søren Marcus Pedersen. Smart Farming Technologies – Description, Taxonomy and Economic Impact. In *Precision Agriculture: Technology and Economic Perspectives*, pages 21–77. Springer, 2017.

[18] Davide Francesco Barbieri, Daniele Braga, Stefano Ceri, Emanuele Della Valle, and Michael Grossniklaus. C-SPARQL: SPARQL for continuous querying. In *18th International Conference on World Wide Web*, pages 1061–1062. ACM, 2009.

[19] Pedro Paulo F. Barcelos, Victor Amorim dos Santos, Freddy Brasileiro Silva, Maxwell E. Monteiro, and Anilton Salles Garcia. An Automated Transformation from OntoUML to OWL and SWRL. *Ontobras*, 1041:130–141, 2013.

[20] Chuda Basnet, Les Foulds, and J. Wilson. Scheduling contractors' farm-to-farm crop harvesting operations. *International Transactions in Operational Research*, 13(1):1–15, 2006.

[21] E. Bastianelli, D. D. Bloisi, R. Capobianco, F. Cossu, G. Gemignani, L. Iocchi, and D. Nardi. On-line semantic mapping. In *16th International Conference on Advanced Robotics*, ICAR, 2013.

[22] Sotiris Batsakis, Euripides G. M. Petrakis, Ilias Tachmazidis, and Grigoris Antoniou. Temporal representation and reasoning in OWL 2. *Semantic Web*, 8(6):981–1000, 2017.

[23] Jan Bauer, Fabian Gehrs, Maximilian Jatzlau, and Stephan Scheuren. Dezentraler Marktplatz in einer offenen, dezentralen Software-Plattform für landwirtschaftliche Dienstleistungen. In *Referate der 38. GIL-Jahrestagung - Digitale Marktplätze und Plattformen*, Lecture Notes in Informatics (LNI), pages 35–38. Gesellschaft für Informatik, 2 2018.

[24] Sean Bechhofer. Owl: Web ontology language. In *Encyclopedia of Database Systems*, pages 2008–2009. Springer, 2009.

[25] M. Beetz, M. Tenorth, and J. Winkler. Open-EASE. In *International Conference on Robotics and Automation*, pages 1983–1990. IEEE, 2015.

[26] Michael Beetz, Ferenc Balint-Benczedi, Nico Blodow, Daniel Nyga, Thiemo Wiedemeyer, and Zoltan-Csaba Marton. RoboSherlock: Unstructured Information Processing for Robot Perception. In *International Conference on Robotics and Automation*. IEEE, 2015.

[27] A. Bernardi. iGreen: Organisationsübergreifendes Wissensmanagement in öffentlichprivater Kooperation. *Automatisierung und Roboter in der Landwirtschaft. KTBL-Tage-2010, Erfurt, Germany*, 2010.

[28] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, 284(5):34–43, 2001.

[29] Heinz Bernhardt. *Schüttguttransport in landwirtschaftlichen Betrieben Deutschlands*. Cuvillier Verlag, 2002.

[30] Heinz Bernhardt, Michael Mederle, Maximilian Treiber, and Sascha Woerz. Aspects of Digization in Agricultural Logistics in Germany. In *46th International Symposium Actual Tasks on Agricultural Engineering*, page 215, 2018.

[31] Gerd Bichler, Wolfgang Matt, and Stefan Engel-Flechsig. Die elektronische Rechnung – Ein Praxisleitfaden für Unternehmen. `http://www.ferd-net.de/upload/Dokumente/Praxisleitfaden_elektronische_Rechnung.pdf`, 2015. Accessed on 01.03.2018.

[32] Simon Blackmore. *The role of yield maps in precision farming.* phdthesis, Cranfield University, 2003.

[33] D. D. Bochtis, P. Dogoulis, Patrizia Busato, C. G. Sørensen, Remigio Berruto, and T. Gemtos. A flow-shop problem formulation of biomass handling operations scheduling. *Computers and Electronics in Agriculture*, 91:49–56, 2013.

[34] D. D. Bochtis, C. G. Sørensen, Patrizia Busato, I. A. Hameed, E. Rodias, Ole Green, and G. Papadakis. Tramline establishment in controlled traffic farming based on operational machinery cost. *Biosystems Engineering*, 107(3):221–231, 2010.

[35] D. D. Bochtis, C. G. Sørensen, and S. G. Vougioukas. Path planning for in-field navigation-aiding of service units. *Computers and Electronics in Agriculture*, 74(1):80–90, 2010.

[36] D. D. Bochtis and Claus G. Sørensen. The vehicle routing problem in field logistics: Part I. *Biosystems Engineering*, 104(4):447–457, 2009.

[37] D. D. Bochtis and Claus G. Sørensen. The vehicle routing problem in field logistics: Part II. *Biosystems Engineering*, 105(2):180–188, 2010.

[38] D. D. Bochtis and S. G. Vougioukas. Minimising the non-working distance travelled by machines operating in a headland field pattern. *Biosystems Engineering*, 101(1):1–12, 2008.

[39] Dionysis Bochtis, Stavros Vougioukas, C. Tsatsarelis, and Y. Ampatzidis. Field operation planning for agricultural vehicles: a hierarchical modeling framework. *Agricultural Engineering International: CIGR Journal*, 9, 2007.

[40] Dionysis D. Bochtis. Machinery management in bio-production systems: planning and scheduling aspects. *Agricultural Engineering International: CIGR Journal*, 12(2):55–63, 2010.

[41] Dionysis D. Bochtis, Claus G. Sørensen, Patrizia Busato, and Remigio Berruto. Benefits from optimal route planning based on B-patterns. *Biosystems Engineering*, 115(4):389–395, 2013.

[42] Dionysis D. Bochtis, Claus G. Sørensen, and Ole Green. A DSS for planning of soil-sensitive field operations. *Decision Support Systems*, 53(1):66–75, 2012.

[43] Dionysis D. Bochtis, Claus G. C. Sørensen, and Patrizia Busato. Advances in agricultural machinery management: A review. *Biosystems Engineering*, 126:69–81, 2014.

[44] Michael D. Boehlje and Vernon R. Eidman. *Farm management*. Wiley, 1984.

[45] Adam Borkowski, Barbara Siemiatkowska, and Jacek Szklarski. Towards Semantic Navigation in Mobile Robotics. *Graph Transformations and Model-driven Engineering*, 5765:719–748, 2010.

[46] André Borrmann and Ernst Rank. Topological operators in a 3D Spatial Query Language for Building Information Models. In *12th International Conference on Computing in Civil and Building Engineering*, ICCCBE, 2008.

[47] André Borrmann and Ernst Rank. Query support for BIMs using semantic and spatial conditions. *Handbook of Research on Building Information Modeling and Construction Informatics: Concepts and Technologies: Concepts and Technologies*, 2009.

[48] André Borrmann and Ernst Rank. Specification and implementation of directional operators in a 3D spatial query language for building information models. *Advanced Engineering Informatics*, 23, 2009.

[49] Abdelbaki Bouguerra, Lars Karlsson, and Alessandro Saffiotti. Monitoring the execution of robot plans using semantic knowledge. *Robotics and Autonomous Systems*, 56(11):942–954, 2008.

[50] Martin Breunig and Sisi Zlatanova. 3D geo-database research: Retrospective and future directions. *Computers and Geosciences*, 37(7):791–803, 2011.

[51] E. Bröhl and Kunibert Mührel. *Transport, Umschlag, Lagerung in der Landwirtschaft*. VEB Verlag Technik, 1983.

[52] Manfred Broy. Cyber-Physical Systems – Wissenschaftliche Herausforderungen bei der Entwicklung. In *Cyber-Physical Systems*, pages 17–31. Springer, 2010.

[53] Bundesverband deutscher Banken e. V. Zahlen, Daten, Fakten der Kreditwirtschaft. `https://bankenverband.de/statistik/banken-deutschland/kreditinstitute-und-bankstellen/#chart-5`, 2017. Accessed on 01.03.2018.

[54] Bundesverband deutscher Banken e.V. Zahlen, Daten, Fakten. `https://bankenverband.de/statistik/banken-deutschland/kreditinstitute-und-bankstellen/#chart-5`, 2018. Accessed on 19.02.2018.

[55] Patrizia Busato, Alessandro Sopegno, Niccolò Pampuro, Luigi Sartori, and Remigio Berruto. Optimisation tool for logistics operations in silage production. *Biosystems Engineering*, 180:146–160, 2019.

[56] Pär Buschka and Alessandro Saffiotti. Some notes on the use of hybrid maps for mobile robots. In *8th International Conference on Intelligent Autonomous Systems*, pages 547–556. CiteSeerX, 2004.

[57] Roberto Capobianco, Guglielmo Gemignani, Domenico Bloisi, Daniele Nardi, and Luca Iocchi. Automatic Extraction of Structural Representations of Environments. In *Intelligent Autonomous Systems*, volume 302 of *Advances in Intelligent Systems and Computing*, pages 721–733. Springer, 2016.

[58] Jeremy J. Carroll, Ian Dickinson, Chris Dollin, Dave Reynolds, Andy Seaborne, and Kevin Wilkinson. Jena: implementing the semantic web recommendations. In *13th International World Wide Web Conference on Alternate Track Papers & Posters*, pages 74–83. Association for Computing Machinery, 2004.

[59] Tsung-Sheng Chang, Yat-wah Wan, and Wei Tsang Ooi. A stochastic dynamic traveling salesman problem with hard time windows. *European Journal of Operational Research*, 198(3):748–759, 2009.

[60] Angelos Chatzimichail, Evangelos Stathopoulos, Dimos Ntioudis, Athina Tsanousa, Maria Rousi, Athanasios Mavropoulos, George Meditskos, Stefanos Vrochidis, and Ioannis Kompatsiaris. Semantic Web and IoT. In *Semantic IoT: Theory and Applications. Studies in Computational Intelligence*, pages 3–33. Springer International Publishing, 2021.

[61] Anna Chlingaryan, Salah Sukkarieh, and Brett Whelan. Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review. *Computers and Electronics in Agriculture*, 151:61–69, 2018.

[62] Konstantinos Christidis and Michael Devetsikiotis. Blockchains and Smart Contracts for the Internet of Things. *IEEE Access*, 4:2292–2303, 2016.

[63] Stefan Christmann, Hendrik Hilpert, Svenja Hagenhoff, and Meik Thöne. Datensicherheit und Datenschutz im Cloud Computing – Risiken und Kriterien zur Anbieterauswahl. *HMD Praxis der Wirtschaftsinformatik*, 47(5):62–70, 2010.

[64] CLAAS. Medien - Download Center. `https://www.claas.de/apps-medien/medien/downloadcenter#`. Accessed on 01.03.2018.

[65] CLAAS KGaA mbH. Website Jaguar Fahrerassistenzsysteme. `https://www.claas.de/produkte/feldhaecksler/jaguar990-930/fahrerassistenzsysteme?subject=webmanager_de_DE`. Accessed on 17.07.2021.

[66] Geoff Clarke and John W. Wright. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4):568–581, 1964.

[67] Eliseo Clementini, Paolino Di Felice, and Peter van Oosterom. A Small Set of Formal Topological Relationships Suitable for End-User Interaction. In *3rd International Symposium on Advances in Spatial Databases*, SSD '93, pages 277–295. Springer, 1993.

[68] Anthony G. Cohn, Brandon Bennett, John Gooday, and Nicholas Mark Gotts. Qualitative Spatial Representation and Reasoning with the Region Connection Calculus. *GeoInformatica*, 1(3):275–316, 1997.

[69] Open Geospatial Consortium. Simple Feature Access - Part 2: SQL Option. http://www.opengeospatial.org/standards/sfs. Accessed on: 13.01.2015.

[70] Silvia Coradeschi and Alessandro Saffiotti. An introduction to the anchoring problem. *Robotics and Autonomous Systems*, 43(2):85–96, 2003.

[71] J. Crespo, R. Barber, and O. M. Mozos. Relational Model for Robotic Semantic Navigation in Indoor Environments. *Journal of Intelligent & Robotic Systems*, 86(3):617–639, 2017.

[72] Gaby G. Dagher, Jordan Mohler, Matea Milojkovic, and Praneeth Babu Marella. Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology. *Sustainable Cities and Society*, 39:283–297, 2018.

[73] Laura Daniele and Luis Ferreira Pires. An ontological approach to logistics. In *Enterprise Interoperability, Research and Applications in the Service-oriented Ecosystem*. ISTE Ltd, John Wiley & Sons, Inc., 2013.

[74] Simon Daum and André Borrmann. Processing of Topological BIM Queries using Boundary Representation Based Methods. *Advanced Engineering Informatics*, 2013.

[75] Charlotte Daydé, Stéphane Couture, Frédérick Garcia, and Roger Martin-Clouaire. Investigating operational decision-making in agriculture. In *7th International Congress on Environmental Modelling and Software*, pages 2188–2195, 2014.

[76] Arne de Wall, Christian Danowski-Buhren, Andreas Wytzisk-Arens, Kai Lingemann, and Santiago Focke Martinez. prospective.HARVEST– Optimizing Planning of Agricultural Harvest Logistic Chains. In *Referate der 40. GIL-Jahrestagung - Informatik in der Land-, Forst- und Ernährungswirtschaft Fokus: Digitalisierung für Mensch, Umwelt und Tier*, Lecture Notes in Informatics (LNI), pages 55–60. Gesellschaft für Informatik, 2020.

[77] H. Deeken, T. Wiemann, K. Lingemann, and J. Hertzberg. SEMAP - a semantic environment mapping framework. In *European Conference on Mobile Robots (ECMR)*, 2015.

[78] Henning Deeken, Sebastian Puetz, Thomas Wiemann, Kai Lingemann, and Joachim Hertzberg. Integrating semantic information in navigational planning. In *41st International Symposium on Robotics*, ISR/Robotik, pages 1–8. VDE, 2014.

[79] Henning Deeken, Thomas Wiemann, and Joachim Hertzberg. A Spatio-Semantic Model for Agricultural Environments and Machines. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 589–600. Springer, 2018.

[80] Deere & Company. Website Active Fill Control. https://www.deere.com/en/technology-products/precision-ag-technology/guidance/active-fill-control/. Accessed on 17.07.2021.

[81] Deutscher Bauernverband e.V. DBV Situationsbericht 2014/ 2015. http://media.repro-mayr.de/43/623743.pdf, 2015. Accessed on 11.04.2019.

[82] Deutsches Forschungszentrum für Künstliche Intelligenz GmbH. iGreen Projekt - agroXML / agroRDF. `http://data.igreen-services.com/`. Accessed on 14.08.2021.

[83] Georg Döring, Andreas Schilcher, Martin Strobl, Roland Schleicher, Martin Seidl, and Johann Mitterleitner. Verfahren zum Transport von Biomasse. techreport, Arbeitsgemeinschaft Landtechnikund landwirtschaftliches Bauwesen in Bayern e.V, 2010.

[84] Brett Drury, Robson Fernandes, Maria-Fernanda Moura, and Alneu de Andrade Lopes. A survey of semantic web technology for agriculture. *Information Processing in Agriculture*, 6(4):487–501, 2019.

[85] Jérôme Dury, Frédérick Garcia, Arnaud Reynaud, and Jacques-Eric Bergez. Cropping-plan decision-making on irrigated crop farms: A spatio-temporal analysis. *European Journal of Agronomy*, 50:1–10, 2013.

[86] Gareth Edwards and Martin Andreas Falk Jensen. Real Time Optimisation of Field Coverage Route Planning with Limited Capacitance Machinery and Spatial Variability. In *XXXV CIOSTA & CIGR V Conference: "From effective to intelligent agriculture and forestry"*, pages 3–5. CIGR International Commission of Agricultural and Biosystems Engineering, CIGR International Commission of Agricultural and Biosystems Engineering, 2013.

[87] Gareth Edwards, Martin Andreas Falk Jensen, and Dionysis D. Bochtis. Coverage planning for capacitated field operations under spatial variability. *International Journal of Sustainable Agricultural Management and Informatics*, 1(2):120–129, 2015.

[88] Gareth T. C. Edwards, Jørgen Hinge, Nick Skou-Nielsen, Andrés Villa-Henriksen, Claus Aage Grøn Sørensen, and Ole Green. Route planning evaluation of a prototype optimised infield route planner for neutral material flow agricultural operations. *Biosystems Engineering*, 153:149–157, 2017.

[89] Max J. Egenhofer and Robert D. Franzosa. Point-set topological spatial relations. *International Journal of Geographical Information System*, 5(2), 1991.

[90] EBICS Electronic Banking Internet Communication Standard. Website. `https://www.ebics.org/about-us/`, 2018. Accessed on 12.02.2018.

[91] Dirk Engelhardt, Nina Zimmemann, and Heinz Bernhardt. Organisation der Getreideernte – Einflussfaktoren und Chancen. In *LAND.TECHNIK 2013: Das Forum für agrartechnische Innovationen*, pages 71–80. VDI Verlag, 2013.

[92] Jacques Ferber and Gerhard Weiss. *Multi-agent systems: an introduction to distributed artificial intelligence*. Addison-Wesley Longman Publishing Co., Inc., 1999.

[93] Dimitris Folinas. A conceptual framework for business intelligence based on activities monitoring systems. *International Journal of Intelligent Enterprise*, 1(1):65–80, 2007.

[94] Food and Agriculture Organization of the United Nations. AGROVOC - Homepage. `http://www.fao.org/agrovoc/home`. Accessed on 14.08.2021.

[95] Spyros Fountas, Giacomo Carli, Claus Grøn Sørensen, Z. Tsiropoulos, Christos Cavalaris, A. Vatsanidou, B. Liakos, Maurizio Canavari, Jens Wiebensohn, and B. Tisserye. Farm management information systems: Current situation and future perspectives. *Computers and Electronics in Agriculture*, 115:40–50, 2015.

[96] Spyros Fountas, Claus G. Sorensen, Zisis Tsiropoulos, Christos Cavalaris, V. Liakos, and T. Gemtos. Farm machinery management information system. *Computers and Electronics in Agriculture*, 110:131–138, 2015.

[97] Spyros Fountas, Dvoralai Wulfsohn, B. Simon Blackmore, H. L. Jacobsen, and Søren Marcus Pedersen. A model of decision-making and information flows for information-intensive agriculture. *Agricultural Systems*, 87(2):192–210, 2006.

[98] Bundesarbeitskreis Futterkonservierung, editor. *Praxishandbuch Futter-und Substratkonservierung*. DLG-Verlag, 2011.

[99] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J. A. Fernandez-Madrigal, and J. Gonzalez. Multi-hierarchical semantic maps for mobile robotics. In *International Conference on Intelligent Robots and Systems*, pages 2278–2283. IEEE, 2005.

[100] Cipriano Galindo, Juan-Antonio Fernández-Madrigal, Javier González, and Alessandro Saffiotti. Robot task planning using semantic maps. *Robotics and Autonomous Systems*, 56(11):955–966, 2008.

[101] Samuel Gan-Mor, Rex L. Clark, and Bruce L. Upchurch. Implement lateral position accuracy under RTK-GPS tractor guidance. *Computers and Electronics in Agriculture*, 59(1-2):31–38, 2007.

[102] Paweł Garbacz and Robert Trypuz. Representation of tensed relations in OWL. In *Research Conference on Metadata and Semantics Research*, pages 62–73. Springer, 2017.

[103] Fred Glover. Tabu search – part I. *ORSA Journal on Computing*, 1(3):190–206, 1989.

[104] David I. Gray, W. J. Parker, and E. Kemp. Farm management research: a discussion of some of the important issues. *Journal of International Farm Management*, 5(1):1–24, 2009.

[105] R. D. Grisso, P. J. Jasa, and D. E. Rolofson. Analysis of traffic patterns and yield monitor data for field efficiency determination. *Applied Engineering in Agriculture*, 18(2):171, 2002.

[106] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[107] Giancarlo Guizzardi. *Ontological foundations for structural conceptual models*. PhD thesis, University of Twente, 2005.

[108] Martin Günther, Thomas Wiemann, Sven Albrecht, and Joachim Hertzberg. Model-based furniture recognition for building semantic object maps. *Artificial Intelligence*, 247:336–351, 2017.

[109] Antonin Guttman. R-trees: A Dynamic Index Structure for Spatial Searching. In *International Conference on Management of Data*, SIGMOD '84, pages 47–57. ACM, 1984.

[110] Mordechai Haklay and Patrick Weber. OpenStreetMap: User-generated Street Maps. *Pervasive Computing*, 7(4):12–18, 2008.

[111] I. A. Hameed, D. D. Bochtis, C. G. Sørensen, and M. Nøremark. Automated generation of guidance lines for operational field planning. *Biosystems Engineering*, 107(4):294–306, 2010.

[112] Georg Happich, Thorsten Lang, and Hans-Heinrich Harms. Modellbasierte Definition des Beladungszustands für parallele Ernte-und Überladeverfahren. In *LAND.TECHNIK 2011: Das Forum für agrartechnische Innovationen*, pages 205–208. VDI, VDI Verlag, Düsseldorf, 2011.

[113] Ian Horrocks, Peter F Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, Mike Dean, et al. Swrl: A semantic web rule language combining owl and ruleml. *W3C Member Submission*, 21(79):1–31, 2004.

[114] Julia Hoxha, Andreas Scheuermann, and Stephan Bloehdorn. An approach to formal and semantic representation of logistics services. In *Workshop on Artificial Intelligence and Logistics (AILog) at 19th European Conference on Artificial Intelligence*, ECAI, pages 73–78, 2010.

[115] International Organization for Standardization. ISO 11783: Tractors and Machinery for Agriculture and Forestry-Serial Control and Communications Data Network. *International Organization for Standardization*, 2011.

[116] Martin Andreas Falk Jensen. Algorithms for operational planning of agricultural field operations. *Technical Reports Mechanical Engineering*, 2(3), 2015.

[117] Martin Andreas Falk Jensen and Dionysis Bochtis. Automatic recognition of operation modes of combines and transport units based on GNSS trajectories. *IFAC Proceedings Volumes*, 46(18):213–218, 2013.

[118] Martin Andreas Falk Jensen, Dionysis Bochtis, Claus Grøn Sørensen, Morten Rufus Blas, and Kasper Lundberg Lykkegaard. In-field and inter-field path planning for agricultural transport units. *Computers & Industrial Engineering*, 63(4):1054–1061, 2012.

[119] Martin F. Jensen, Dionysis Bochtis, and Claus G. Sørensen. Coverage planning for capacitated field operations, part II: Optimisation. *Biosystems Engineering*, 139:149–164, 2015.

[120] Martin F. Jensen, Michael Nørremark, Patrizia Busato, Claus G. Sørensen, and Dionysis Bochtis. Coverage planning for capacitated field operations, Part I: Task decomposition. *Biosystems Engineering*, 139:136–148, 2015.

[121] Alexandros Kaloxylos, Aggelos Groumas, Vassilis Sarris, Lampros Katsikas, Panagis Magdalinos, Eleni Antoniou, Zoi Politopoulou, Sjaak Wolfert, Christopher Brewster, Robert Eigenmann, et al. A cloud-based Farm Management System: Architecture and implementation. *Computers and Electronics in Agriculture*, 100:168–179, 2014.

[122] Michio Kise, Noboru Noguchi, Kazunobu Ishii, and Hideo Terao. Enhancement of turning accuracy by path planning for robot tractor. In *Conference on Automation Technology for Off-Road Equipment*. American Society of Agricultural and Biological Engineers, 2002.

[123] Florian Klingler, Johannes Blobel, and Falko Dressler. Agriculture meets IEEE 802.11 p: a feasibility study. In *15th International Symposium on Wireless Communication Systems*, ISWCS, pages 1–6. IEEE, IEEE, 2018.

[124] P. Klink, C. Mertens, and K. Kompalka. ERP-Marktstudie 2016: Auswirkungen von Industrie 4.0 auf die Anforderungen an ERP-Systeme. `http://publica.fraunhofer.de/dokumente/N-438257.html`, 2018. Accessed on 10.03.2018.

[125] Andre Kluge. CLAAS Fleet View – Mobile application for coordinating the transport logistics in the grain harvest. In *LAND.TECHNIK 2015: Das Forum für agrartechnische Innovationen*, pages 355–358. VDI Verlag, 2015.

[126] Bruno Koch. Die E-Rechnung steht im Zeichen großer Marktveränderungen. `www.billentis.com/marktreport_e-rechnung_DE.htm`, 2017. Accessed on 01.03.2018.

[127] Ahmed Kosba, Andrew Miller, Elaine Shi, Zikai Wen, and Charalampos Papamanthou. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *Symposium on Security and Privacy*, pages 839–858. IEEE, 2016.

[128] Ioannis Kostavelis and Antonios Gasteratos. Semantic mapping for mobile robotics tasks: A survey. *Robotics and Autonomous Systems*, 66:86–103, 2015.

[129] Franz Kraatz, Frank Nordemann, and Ralf Tönjes. Anbindung von ISOBUS-Geräten an ein Online Precision Farming System. In *Referate der 35. GIL-Jahrestagung - Komplexität versus Bedienbarkeit / Mensch-Maschine-Schnittstellen*, Lecture Notes in Informatics (LNI), pages 89–92. Gesellschaft für Informatik in der Land-, Forst- und Ernährungswirtschaft (GIL), Gesellschaft für Informatik, 2015.

[130] Florian Krampe, Henning Deeken, Thilo Steckel, and Nils Aschenbruck. HOAR: A hybrid-opportunistic architecture for robust agricultural networking. In *IoT Vertical and Topical Summit on Agriculture*, pages 1–6. IEEE, 2018.

[131] Holger Kremer and Clemens Westerkamp. Context aware decision support for mobile participants of distributed production processes. In *17th International Conference on Emerging Technologies & Factory Automation*, ETFA, pages 1–8. IEEE, IEEE, 2012.

[132] M. Krouik, Z. Kviz, F. Kumhala, J. Hula, and T. Loch. Procedures of soil farming allowing reduction of compaction. *Precision Agriculture*, 12(3):317–333, 2011.

[133] Kuratorium für Technik und Bauwesen in der Landwirtschaft e. V. KTBL - agroXML / agroRDF. `https://vest.agrisemantics.org/content/agroxml-agrordf`. Accessed on 14.08.2021.

[134] Kuratorium für Technik und Bauwesen in der Landwirtschaft e. V. KTBL - Homepage. `https://www.ktbl.de/`. Accessed on 14.08.2021.

[135] Christian Landsiedel, Verena Rieser, Matthew Walter, and Dirk Wollherr. A review of spatial reasoning and interaction for real-world robotics. *Advanced Robotics*, 31(5):222–242, 2017.

[136] Dagmar Lang and Dietrich Paulus. Semantic Maps for Robotics. In *Workshop on AI Robotics at ICRA2014*, ICRA. IEEE, 2014.

[137] Johannes Lauer, Ludwig Richter, Timothy Ellersiek, and Alexander Zipf. TeleAgro+: Analysis Framework for Agricultural Telematics Data. In *7th ACM SIGSPATIAL International Workshop on Computational Transportation Science*, IWCTS '14, pages 47–53. ACM, 2014.

[138] Lartey G. Lawson, Søren Marcus Pedersen, Claus Grøn Sørensen, Lisa Pesonen, Spyros Fountas, Armin Werner, Frank W. Oudshoorn, Luzia Herold, Thanos Chatzinikos, and Inger Marie Kirketerp. A four nation survey of farm information management and advanced farming systems: A descriptive analysis of survey responses. *Computers and Electronics in Agriculture*, 77(1):7–20, 2011.

[139] Tony Lewis. Evolution of farm management information systems. *Computers and Electronics in Agriculture*, 19(3):233–248, 1998.

[140] Klaus Liebler, Ulrike Beissert, Marco Motta, and Axel Wagenitz. Introduction OTD-Net and LAS: Order-to-delivery network simulation and decision support systems in complex production and logistics networks. In *2013 Winter Simulations Conference (WSC)*, pages 439–451. IEEE, 2013.

[141] Tyler Mark, Terry Griffin, et al. Defining the Barriers to Telematics for Precision Agriculture: Connectivity Supply and Demand. In *2016 Annual Meeting of the Southern Agricultural Economics Association*. Southern Agricultural Economics Association, 2016.

[142] Tyler B. Mark, Brian Whitacre, and Terry Griffin. Assessing the Value of Broadband Connectivity for Big Data and Telematics: Technical Efficiency. In *2015 Annual Meeting of the Southern Agricultural Economics Association*. Southern Agricultural Economics Association, 2015.

[143] Roger Martin-Clouaire and Jean-Pierre Rellier. Modelling and simulating work practices in agriculture. *International Journal of Metadata Semantics and Ontologies*, 4(1):42–53, 2009.

[144] Roger Martin-Clouaire and Jean-Pierre Rellier. Dynamic resource allocation in farm management simulation. *19th International Congress on Modelling and Simulation*, pages 808–814, 2011.

[145] Daniel Martini, Mario Schmitz, and Esther Mietzsch. agroRDF as a Semantic Overlay to agroXML: a General Model for Enhancing Interoperability in Agrifood Data Standards. In *CIGR Conference on Sustainable Agriculture through ICT Innovation*, 2013.

[146] Aline Freitas Martins and Ricardo de Almeida Falbo. Models for Representing Task Ontologies. In *3rd Workshop on Ontologies and their Applications*, volume 427 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2008.

[147] Brian McBride. Jena: A semantic web toolkit. *IEEE Internet Computing*, 6(6):55–59, 2002.

[148] Edson Murakami, Antonio M. Saraiva, Luiz C. M. Ribeiro Junior, Carlos E. Cugnasca, Andre R. Hirakawa, and Pedro L. P. Correa. An infrastructure for the development of distributed service-oriented information systems for precision agriculture. *Computers and Electronics in Agriculture*, 58(1):37–48, 2007.

[149] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, 2008.

[150] Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder. *Bitcoin and cryptocurrency technologies: a comprehensive introduction*. Princeton University Press, 2016.

[151] Raimo Nikkilä, Ilkka Seilonen, and Kari Koskinen. Software architecture for farm management information systems in precision agriculture. *Computers and Electronics in Agriculture*, 70(2):328–336, 2010.

[152] Andreas Nüchter and Joachim Hertzberg. Towards semantic maps for mobile robots. *Robotics and Autonomous Systems*, 56(11):915–926, 2008.

[153] P. L. Nuthall. Managerial ability – a review of its basis and potential improvement using psychological concepts. *Agricultural Economics*, 24(3):247–262, 2001.

[154] Timo Oksanen, Raimo Linkolehto, and Ilkka Seilonen. Adapting an industrial automation protocol to remote monitoring of mobile agricultural machinery: a combine harvester with IoT. In *IFAC Conference on Sensing, Control and Automation Technologies for Agriculture*, volume 49, pages 127–131. IFAC, Elsevier, 2016.

[155] OntoUML Community. OntoUML - Homepage. `https://ontouml.org/`. Accessed on 14.08.2021.

[156] A. Orfanou, P. Busato, D. D. Bochtis, G. Edwards, D. Pavlou, C. G. Sørensen, and R. Berruto. Scheduling for machinery fleets in biomass multiple-field operations. *Computers and Electronics in Agriculture*, 94:12–19, 2013.

[157] OSGeo. GEOS library. `http://trac.osgeo.org/geos/`. Accessed on: 13.01.2015.

[158] Oslandia. SFCGAL Project Website. `http://www.sfcgal.org/`. Accessed on: 13.01.2015.

[159] R. J. Palmer, D. Wild, and K. Runtz. Improving the Efficiency of Field Operations. *Biosystems Engineering*, 84(3):283–288, 2003.

[160] Dejan Pangercic, Benjamin Pitzer, Moritz Tenorth, and Michael Beetz. Semantic object maps for robotic housework-representation, acquisition and use. In *International Conference on Intelligent Robots and Systems*, pages 4644–4651. IEEE, 2012.

[161] Matthias Parlings and Katharina Kompalka. Digital Business Transformation unter Berücksichtigung der Möglichkeiten und Restriktionen vorhandener ERP-Systeme. *Automation & Digitalisierung Web Magazin*, 2017.

[162] Søren Marcus Pedersen and K. M. Lind. Precision Agriculture – From Mapping to Site-Specific Application. In *Precision Agriculture: Technology and Economic Perspectives*, pages 1–20. Springer, 2017.

[163] Søren Marcus Pedersen and Kim Martin Lind, editors. *Precision Agriculture: Technology and Economic Perspectives*. Springer, 2017.

[164] Dohn Pfeiffer and Sebastian Blank. Real-time operator performance analysis in agricultural equipment. In *73rd International Conference on Agricultural Engineering (AgEng)*, pages 6–7, 2015.

[165] Francis J. Pierce and Peter Nowak. Aspects of Precision Agriculture. In *Advances in Agronomy*, volume 67, pages 1–85. Elsevier, 1999.

[166] Victor Pillac, Michel Gendreau, Christelle Guéret, and Andrés L. Medaglia. A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1):1–11, 2013.

[167] PostGIS Project Steering Committee. PostGIS - Website. `http://www.postgis.net/`. Accessed on 03.02.2015.

[168] PostGIS Project Steering Committee. PostGIS Website. `http://postgis.com/`. Accessed on: 13.01.2015.

[169] Project SOFiA. Website. `http://www.sofia-projekt.de`, 2018. Accessed on 01.03.2018.

[170] A. Pronobis, B. Caputo, P. Jensfelt, and H. I. Christensen. A realistic benchmark for visual indoor place recognition. *Robotics and Autonomous Systems*, 58(1):81–96, 2010.

[171] Andrzej Pronobis and Patric Jensfelt. Large-scale semantic mapping and reasoning with heterogeneous modalities. In *International Conference on Robotics and Automation*, pages 3515–3522. IEEE, 2012.

[172] Julian Quindt, Eike Reetz, Vanessa Kukuck, Ralf Tönjes, and Clemens Westerkamp. Agent based decision support system for optimizing logistical processes in agricultural production. In *9th International Conference on Industrial Informatics*, pages 27–32. IEEE, IEEE, 2011.

[173] Jannik Redenius and Matthias Dingwerth. A multilevel simulation framework for highly automated harvest processes enabled by environmental sensor systems. In *LAND.TECHNIK 2017: Das Forum für agrartechnische Innovationen*, pages 157–163. VDI Verlag, 2017.

[174] Jochen Renz, Bernhard Nebel, and Ian Pratt-Hartmann. Qualitative Spatial Reasoning Using Constraint Calculi. In *Handbook of Spatial Logics*, pages 161–215. Springer, 2007.

[175] P. Reyns, B. Missotten, H. Ramon, and J. De Baerdemaeker. A review of combine sensors for precision farming. *Precision Agriculture*, 3(2):169–182, 2002.

[176] James Risbey, Milind Kandlikar, Hadi Dowlatabadi, and Dean Graetz. Scale, context, and decision making in agricultural adaptation to climate variability and change. *Mitigation and Adaptation Strategies for Global Change*, 4(2):137–165, 1999.

[177] Marion Robert, Alban Thomas, and Jacques-Eric Bergez. Processes of adaptation in farm decision-making models. a review. *Agronomy for Sustainable Development*, 36(4):64, 2016.

[178] David C. Rose, William J. Sutherland, Caroline Parker, Matt Lobley, Michael Winter, Carol Morris, Susan Twining, Charles Ffoulkes, Tatsuya Amano, and Lynn V. Dicks. Decision support tools for agriculture: Towards effective design and delivery. *Agricultural Systems*, 149:165–174, 2016.

[179] J. R. Ruiz-Sarmiento, Cipriano Galindo, and Javier González-Jiménez. Robot@Home, a Robotic Dataset for Semantic Mapping of Home Environments. *International Journal of Robotics Research*, 2017.

[180] Stephan Scheuren. Process-Optimized Planning for Cooperative Mobile Robots. *KI-Künstliche Intelligenz*, 28(4):325–328, 2014.

[181] Stephan Scheuren. *Prozessoptimierte Planung für kooperative mobile Roboter*. phdthesis, Osnabrück University, 2014.

[182] Stephan Scheuren, Stefan Stiene, Ronny Hartanto, Joachim Hertzberg, and Max Reinecke. Spatio-temporally constrained planning for cooperative vehicles in a harvesting scenario. *KI-Künstliche Intelligenz*, 27(4):341–346, 2013.

[183] Daniela Schmidt, Rafael H. Bordini, Felipe Meneguzzi, and Renata Vieira. An Ontology for Collaborative Tasks in Multi-agent Systems. In *Ontobras*, volume 1442, 2015.

[184] Julian Schütte, Gilbert Fridgen, Wolfgang Prinz, Thomas Rose, Nils Urbach, Thomas Hoeren, Nikolas Guggenberger, Christian Welzel, Steffen Holly, Axel Schulte, Philipp Sprenger, Christian Schwede, Birgit Weimert, Boris Otto, Mathias Dalheimer, Markus Wenzel, Michael Kreutzer, Michael Fritz, Ulrich Leiner, and Alexander Nouak. Blockchain und Smart Contracts : Technologien, Forschungsfragen und Anwendungen. `https://eref.uni-bayreuth.de/43980/`, 2017. Accessed on 01.03.2018.

[185] Hasan Seyyedhasani and Joseph S. Dvorak. Using the Vehicle Routing Problem to reduce field completion times with multiple machines. *Computers and Electronics in Agriculture*, 134:142–150, 2017.

[186] Hasan Seyyedhasani and Joseph S. Dvorak. Dynamic rerouting of a fleet of vehicles in agricultural operations through a Dynamic Multiple Depot Vehicle Routing Problem representation. *Biosystems Engineering*, 171:63–77, 2018.

[187] Hasan Seyyedhasani and Joseph S. Dvorak. Reducing field work time using fleet routing optimization. *Biosystems Engineering*, 169:1–10, 2018.

[188] Theodore James Shinners, Matthew Francis Digman, and John C. Panuska. Overlap Loss of manually and Automatically Guided Mowers. *Applied Engineering in Agriculture*, 28(1):5–8, 2010.

[189] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor Segmentation and Support Inference from RGBD Images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012.

[190] K. Sjöö, A. Pronobis, and P. Jensfelt. Functional topological relations for qualitative spatial representation. In *15th International Conference on Advanced Robotics*, ICAR, pages 130–136, 2011.

[191] Henning T. Søgaard and Claus G. Sørensen. A model for optimal selection of machinery sizes within the farm machinery system. *Biosystems Engineering*, 89(1):13–28, 2004.

[192] C. Solano, H. León, E. Pérez, Lise Tole, R. H. Fawcett, and Mario Herrero. Using farmer decision-making profiles and managerial capacity as predictors of farm management and performance in Costa Rican dairy farms. *Agricultural Systems*, 88(2-3):395–428, 2006.

[193] C. G. Sørensen, S. Fountas, E. Nash, Liisa Pesonen, Dionysis Bochtis, Søren Marcus Pedersen, B. Basso, and S. B. Blackmore. Conceptual model of a future farm management information system. *Computers and Electronics in Agriculture*, 72(1):37–47, 2010.

[194] C. G. Sørensen, S. Fountas, E. Nash, Liisa Pesonen, Dionysis Bochtis, Søren Marcus Pedersen, B. Basso, and S. B. Blackmore. Conceptual model of a future farm management information system. *Computers and Electronics in Agriculture*, 72(1):37–47, 2010.

[195] C. G. Sorensen, Lisa Pesonen, S. Fountas, Pasi Suomi, D. Bochtis, P. Bildsøe, and S. M. Pedersen. A user-centric approach for information modelling in arable farming. *Computers and Electronics in Agriculture*, 73(1):44–55, 2010.

[196] Claus G. Sørensen and Dionysis D. Bochtis. Conceptual Model of Fleet Management in Agriculture. *Biosystems Engineering*, 105(1):41–50, 2010.

[197] Claus G. Sørensen and Villy Nielsen. Operational analyses and model comparison of machinery systems for reduced tillage. *Biosystems Engineering*, 92(2):143–155, 2005.

[198] Claus G. Sørensen, Efthymios Rodias, and Dionysis Bochtis. Auto-Steering and Controlled Traffic Farming – Route Planning and Economics. In *Precision Agriculture: Technology and Economic Perspectives*, pages 129–145. Springer, 2017.

[199] Mark Spekken and Sytze de Bruin. Optimized routing on agricultural fields by minimizing maneuvering and servicing time. *Precision Agriculture*, 14(2):224–244, 2013.

[200] John V. Stafford. Implementing precision agriculture in the 21st century. *Journal of Agricultural Engineering Research*, 76(3):267–275, 2000.

[201] Thilo Steckel. *Entwicklung einer kontextbasierten Systemarchitektur zur Verbesserung des kooperativen Einsatzes mobiler Arbeitsmaschinen.* Dissertation, Universität Hohenheim, 2018.

[202] Georg Steinberger, Matthias Rothmund, and Hermann Auernhammer. Mobile farm equipment as a data source in an agricultural service architecture. *Computers and Electronics in Agriculture*, 65(2):238–246, 2009.

[203] Sebastian Stock, Kai Lingemann, Stefan Stiene, and Joachim Hertzberg. Towards a flexible hybrid planner for machine coordination in arable farming. In *Referate der 36. GIL-Jahrestagung - Intelligente Systeme - Stand der Technik und neue Möglichkeiten*, pages 205–208. Gesellschaft für Informatik e.V., 2016.

[204] Melanie Swan. *Blockchain: Blueprint for a new economy.* O'Reilly Media, Inc., 2015.

[205] R. Sylvester-Bradley, E. Lord, D. L. Sparkes, R. K. Scott, J. J. J. Wiltshire, and J. Orson. An analysis of the potential of Precision Farming in Northern Europe. *Soil Use and Management*, 15(1):1–8, 1999.

[206] Michael Szydlo. Merkle tree traversal in log space and time. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 541–554. Springer, 2004.

[207] Randal K. Taylor, Mark D. Schrock, and Scott A. Staggenborg. Extracting machinery management information from GPS data. In *2002 ASAE Annual Meeting*, page 1. American Society of Agricultural and Biological Engineers, American Society of Agricultural and Biological Engineers, 2002.

[208] Moritz Tenorth and Michael Beetz. KnowRob – A Knowledge Processing Infrastructure for Cognition-enabled Robots. *International Journal of Robotics Research*, 32(5):566–590, 2013.

[209] Moritz Tenorth, Stefan Profanter, Ferenc Balint-Benczedi, and Michael Beetz. Decomposing CAD models of objects of daily use and reasoning about their functional parts. *International Conference on Intelligent Robots and Systems*, 2013.

[210] The Apache Software Foundation. Apache Jena - Website. `https://jena.apache.org/`. Accessed on 14.08.2021.

[211] The Berlin Group. Website. `https://www.berlin-group.org/`, 2018. Accessed on 01.03.2018.

[212] The CGAL Project. Computational Geometry Algorithms Library (CGAL). `http://cgal.com/documentation`. Accessed on: 13.01.2015.

[213] S. Tomic, D. Drenjanac, G. Lazendic, S. Hörmann, F. Handler, W. Wöber, K. Schulmeister, M. Otte, and W. Auer. agriOpenLink: Semantic Services for Adaptive Processes in Livestock Farming. In *International Conference of Agricultural Engineering*, 2014.

[214] Slobodanka Dana Kathrin Tomic, Anna Fensel, Christian Aschauer, Klemens Gregor Schulmeister, Thomas Riegler, Franz Handler, Marcel Otte, and Wolfgang Auer. agriOpenLink: Towards adaptive agricultural processes enabled by open interfaces, linked data and services. In *Research Conference on Metadata and Semantic Research*, pages 408–413. Springer, 2013.

[215] Paolo Toth and Daniele Vigo. *The Vehicle Routing Problem*. SIAM, 2002.

[216] Naoum K. Tsolakis, Christos A. Keramydas, Agorasti K. Toka, Dimitrios A. Aidonis, and Eleftherios T. Lakovou. Agrifood Supply Chain Management: A comprehensive hierarchical decision-making framework and a critical taxonomy. *Biosystems Engineering*, 120:47–64, 2014.

[217] Alan Mathison Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 2(1):230–265, 1937.

[218] Frits K. van Evert, Paul van der Voet, Eric van Valkengoed, Lammert Kooistra, and Corné Kempenaar. Satellite-based herbicide rate recommendation for potato haulm killing. *European Journal of Agronomy*, 43:49–57, 2012.

[219] S. Vougioukas and N. Sigrimis. Coordinated path tracking for teams of agricultural robotic vehicles. *IFAC AgriControl*, pages 165–170, 2007.

[220] Harald Vranken. Sustainability of Bitcoin and Blockchains. *Current Opinion in Environmental Sustainability*, 28:1–9, 2017.

[221] Markus Waibel, Michael Beetz, Javier Civera, Raffaello d'Andrea, Jos Elfring, Dorian Galvez-Lopez, Kai Haussermann, Rob Janssen, J. M. M. Montiel, and Alexander Perzylo. A world wide web for robots. *IEEE Robotics & Automation Magazine*, 18(2):69–82, 2011.

[222] Cornelia Weltzien. *Assistenzsystem für den Überladevorgang bei einem selbstfahrenden Feldhäcksler*. Shaker, 2009.

[223] T. Wiemann, K. Lingemann, A. Nüchter, and J. Hertzberg. A Toolkit for Automatic Generation of Polygonal Maps – Las Vegas Reconstruction. In *ROBOTIK 2012*. VDE Verlag, 2012.

[224] Markus Witthaut. Supply-Chain-Management-Systeme. In *IT in der Logistik 2013/2014*, pages 144–160. Fraunhofer Verlag, 2013.

[225] Markus Witthaut, Marcel David, and Philipp Klink. Smart Objects. Dezentrale Planung und Steuerung von Supply Chains. *Jahrbuch Logistik*, 2017:114–119, 2017.

[226] Markus Witthaut, Henning Deeken, Philipp Sprenger, Petyo Gadzhanov, and Marcel David. Smart Objects and Smart Finance for Supply Chain Management. *Logistics Journal: referierte Veröffentlichungen*, 2017(10), 2017.

[227] Wilfried Wöber, Klemens Gregor Schulmeister, Christian Aschauer, Andreas Gronauer, Dana Kathrin Tomic, Anna Fensel, Thomas Riegler, Franz Handler, Sandra Hörmann, Marcel Otte, and Wolfgang Auer. agriOpenLink: Adaptive Agricultural Processes via Open Interfaces and Linked Services. In *IT-Standards in der Agrar- und Ernährungswirtschaft – Fokus: Risiko- und Krisenmanagement*, pages 157–160, Bonn, 2014. Gesellschaft für Informatik e.V.

[228] Diedrich Wolter and Jan Wallgrün. Qualitative spatial reasoning for applications: New challenges and the SparQ toolbox. *Qualitative Spatio-Temporal Representation and Reasoning: Trends and Future Directions*, pages 336–362, 2012.

[229] World Wide Web Consortium. RDF. `https://www.w3.org/RDF/`. Accessed on 14.08.2021.

[230] World Wide Web Consortium. Semantic Web. `https://www.w3.org/standards/semanticweb/`. Accessed on 14.08.2021.

[231] World Wide Web Consortium. SPARQL Query Language for RDF. `https://www.w3.org/TR/rdf-sparql-query/`. Accessed on 11.04.2019.

[232] J. Wu, H. I. Christensen, and J. M. Rehg. Visual Place Categorization: Problem, dataset, and algorithm. In *International Conference on Intelligent Robots and Systems*, pages 4763–4770. IEEE, 2009.

[233] Jin Xu, Jihua Meng, and Lindi J. Quackenbush. Use of remote sensing to predict the optimal harvest date of corn. *Field Crops Research*, 236, 2019.

[234] Naiqian Zhang, Maohua Wang, and Ning Wang. Precision Agriculture – A Worldwide Overview. *Computers and Electronics in Agriculture*, 36(2-3):113–132, 2002.

[235] Siyka Zlatanova, Alias Abdul Rahman, and Wenzhong Shi. Topological models and frameworks for 3D spatial objects. *Computers & Geosciences*, 30(4):419–428, 2004.

Appendix A

Appendix

A.1 List of Abbreviations

AC Agricultural Contractor

ADS Autonomous Decision System

AgriCo Agricultural Logistics Core Ontology

AgriServ Agricultural Logistics Service Ontology

AGS Automatic Guidance System

APS Automated Platooning System

B2B Business-to-Business

B2C Business-to-Customer

BGA Biogasanlage

BIC Business Identifier Code

BGP biogas plant

BMBF Bundesministerium für Bildung und Forschung

CAN Controller Area Network

CPS Cyber-Physical System

CTF Controlled Traffic Farming

CV Compactor Vehicle

DL Description Logic

DSS Decision Support System

EBICS Electronic Banking Communication Standard

ECU Electronic Control Unit

EDI Electronic Data Interchange

EDP Electronic Data Processing

ERP Enterprise Resource Planning

FAO Food and Agriculture Organization of the United Nations

FeRD Forum elektronische Rechnung Deutschland

FinTS Financial Transaction Services

FMIS Farm Management Information System

FMS Fleet Management System

GIS Geographic Information System

GNSS Global Navigation Satellite System

GPS Global Positioning System

GUI Graphical User Interface

IBAN International Bank Account Number

IML Fraunhofer Institut für Materialfluss und Logistik

IoT Internet of Things

IS Information System

ISS Information Support System

KPI Key Performance Indicator

KTBL Kuratorium für Technik und Bauwesen in der Landwirtschaft e. V.

LiDAR Light Detection And Ranging

LOD Linked Open Data

LogiCo Logistics Core Ontology

LogiServ Logistics Service Ontology

LU Lohnunternehmer

LW Landwirt

M2M Machine-to-Machine

MIS Management Information System

NDVI Normalized Difference Vegetation Index

OGC Open Geospatial Consortium

OPA Online Process Analytics

OPS Online Process Support

OWL Web Ontology Language

PF Precision Farming

PFA Precision Farming Application

PoI Point of Interest

PSD2 Payment Service Directive 2

PTO Power Take-Off

RDF Resource Description Framework

ReitFeRD Report Invoice by Template with ZUGFeRD

RoI Region of Interest

ROS Robot Operating System

RTK Real-time Kinematic

SCP Smart Contract Platform

SEMAP Semantic Environment Mapping Framework

SFH Self-propelled Forage Harvester

SO Smart Object

SOFiA Prozessinnovation in Planung und Steuerung von Wertschöpfungsnetzwerken durch Integration von Smart Objects und Smart Finance Ansätzen

SPARQL SPARQL Protocol and RDF Query Language

SQL Structured Query Language

SWRL Semantic Web Rule Language

TV Transport Vehicle

UAV Unmanned Aerial Vehicle

UFO Unified Foundation Ontology

VRA Variable Rate Application

VRP Vehicle Routing Problem

W3C World Wide Web Consortium

XML Extensible Markup Language

ZUGFeRD Zentraler User-Guide des Forum elektronische Rechnung Deutschland (FeRD)

Proclamation

Hereby I confirm that I wrote this thesis independently and that I have not made use of any other resources or means than those indicated.

Osnabrück, August 2021